

CLUB I.S.I.S.

SPECTRAVIDEO™

AÑO - 2 - NUMERO - 5 - JUNIO - JULIO - AGOSTO - SEPBRE 1986

PROMUEVE Y EDITA
I.S.I.S. s.a. I.S.I.S.
c/ Lagasca,125
Tlf. 411 55 61
411 55 43
28006 - MADRID

Imprime ARTYDIS
c/ Gabriel Lobo,10
Tlf. 411 44 10
28002 - MADRID

D.L.M-165-1985

INDICE

COLABORAN

J.CARLOS AMOROS
C.GARCIA DOS SANTOS
J.CARLOS BUENO
PABLO YU
MARIVI ARRAZOLA

Estimados Socios:

Cada vez resulta más difícil completar un boletín. Por este motivo, no nos podemos comprometer a comenzar un nuevo año manteniendo la obligación de publicarlo.

Después de pulsar la opinión de bastantes compañeros, la mayoría es partidaria de continuar con el CLUB. Quizás como reconocimiento del apoyo que han tenido, o porque existen lazos de unión importantes de los que se han derivado ventajas económicas, aplicaciones, información, etc. Por lo tanto, una vez más nos abrimos a vuestras sugerencias. A nosotros tampoco nos satisface romper el cordón umbilical. Por eso proponemos lo siguiente:

- Una cuota, por ejemplo de 2.000,- Pts., para poder atender las dudas, publicar boletines cuando exista material suficiente, enviar datos actualizados sobre los productos más interesantes, conseguir mejores descuentos para vosotros y vuestros amigos.... En fin, ayudar procurando mantener el espíritu de amistad y colaboración, que directa o indirectamente, beneficia a todos.

SVI - ISIS

Como os deciamos al principio, la gama de productos SPECTRAVIDEO es más amplia de lo que pensábamos. Por si alguien tiene interés en ellos, consideramos interesante que los conozcáis.

- Compacto SVI 25 w por canal. Doble pletina. Alta velocidad de grabación. Radio AM-FM. Plato. Ecualizador. Compact-Disk (Rayo Laser). Cajas acústicas de 2 vías. Mueble Rack.

Precio: 90.000,-Pts.

- Compacto SVI 25 w por canal. Doble pletina. Alta velocidad de grabación. Radio AF-FM. Plato. Ecualizador. Mando a distancia por sistema infrarrojo. Compact-Disk (Rayo Laser). Dos cajas acústicas de 3 vías. Mueble Rack.

Precio:108.000,- "

- Compact - Disk (Rayo Laser)

Precio: 45.000,- "

- Compacto SVI 25 w por canal. Doble pletina. Alta velocidad de grabación. Radio AM-FM. Plato. Ecualizador. Mando a distancia por sistema infrarrojo. Dos cajas acústicas de 2 vías. Mueble Rack.

Precio: 58.500,- "

- Compacto SVI 25 w por canal. Doble pletina. Alta velocidad de grabación. Radio AM_FM. Plato. Ecualizador. Dos cajas acústicas de 2 vías. Mueble Rack.

Precio: 54.000,- "

A LOS INTERESADOS LES ENVIAREMOS UN FOLLETO CON LAS CARACTERISTICAS, PORQUE ESTE TEMA NO ES NUESTRO FUERTE.

PC

En vista del interés demostrado por muchos socios con respecto a los Compatibles, incluimos en el boletín estos artículos.

COMPATIBILIDAD PC's

Cuando IBM anunció su Personal Computer, causaron impacto las siguientes novedades:

- Incorporación de 16 bits a la Memoria Principal.
- Sistema Operativo MS DOS. Aunque análogo al CP/M rompía con su hegemonía y con la de los microordenadores de 8 bits.

Inicialmente el software disponible, al menos al comienzo de su introducción en España, fué muy pobre. Razón por la cual tardó en imponerse en el mercado y hubo por por este motivo muchos clientes descontentos. También es verdad que se compran los microordenadores sin tener en cuenta que son pura chatarra, cuando la inteligencia del hombre falta o falla.

Las siglas de IBM lograron que muchísimas empresas de Analistas y Programadores, dotaran a los PC's del software más completo entre todos los micros. Por lo cual han acabado imponiéndose, hasta tal punto, que ya no se puede aconsejar a cualquier empresa micros que no sean PC's. Han barrido el mercado, desplazando a todos sus competidores.

El brillante futuro de los PC's fué intuído por muchas Compañías, la mayoría de ellas desconocidas, que lograron imitarlo. Nacen los COMPATIBLES. Su éxito ha sido tan rotundo que ha obligado a todos los grandes fabricantes a sacar sus compatibles para poder estar presentes en este mercado.

Se plantea el grado de compatibilidad en función de que la imitación sea mejor o peor, e incluso mejorada, porque algunos han querido ponerle la guinda. Sin entrar en detalles de estas guindas, útiles la mayoría de las veces para cubrir necesidades muy específicas, la gestión, es decir, la utilización de los PC's para resolver Aplicaciones tradicionales de Contabilidad, Facturación, Control de Almacén, etc., se llevan la palma de unidades vendidas.

Al haber bajado tanto el precio de los Compatibles, se desata la fiebre de utilizar el grado de compatibilidad como argumento comercial. Aunque puede calificarse en la actualidad como excelente, sobre todo en las Aplicaciones más corrientes (Contabilidades, Facturaciones,-- DBASE II, DBASE III...), las principales dificultades provienen de las desarrolladas con el BASICA. Este lenguaje es específico de IBM y emplea elementos de la ROM BIOS que no ha sido totalmente imitada por poseer IBM los derechos absolutos. En una prueba realizada con el Bondwell, los programas en Basica corrían bien cuando-- no utilizaban rutinas o instrucciones de dicha ROM. En caso contrario el Sistema se quedaba colgado, por no-hallarla en su memoria. Esto les ocurre sin excepción- a todos los compatibles. No existe la compatibilidad -- 100 % absoluta. (A no ser que se piratee dicha información y ROM que está legalmente prohibido).

Afortunadamente, la inmensa mayoría de las Aplicaciones disponibles no han sido desarrolladas en BASICA, por lo tanto el intercambio de software entre los compatibles es el pan nuestro de cada día. Los compatibles han permitido enriquecer mucho más las posibilidades de utilización de los Personal Computer y reducir considerablemente su costo.

- PLOTTER IMP (4 colores) Serie y Paralelo .... 157.500,- PTAS.
- PLOTTER PLANO (4 colores) Serie y Paralelo .. 210.000,- "
- Monitor 12" fósforo verde, con sonido .... 21.000,- "
- Monitor 12" Orientable .... 27.000,- "
- Monitor 12" Para Compatibles .... 19.900,- "

- Tarjeta Color Gráficos PC y AT .... 14.000,- "
- Tarjeta Color Gráficos Printer PC y AT ...... 18.000,- "
- Tarjeta Hercules + Printer PC y AT .... 29.900,- "

Para cualquier duda y otras posibilidades, llamar al CLUB.

SOFTWARE DE APLICACION PARA PC's

- Contabilidad General PLACON .... 30.000,- PTAS.
- Módulo Liquidación IVA .... 15.000,- "
- Facturación Alfa/3 .... 34.000,- "
- Alfa/3 y Placon .... 60.000,- "
- Gestión del Almacén .... 25.000,- "
- Facturación y Contabilidad General para Talleres Repar. de Automóviles .... 125.000,- "
- Contabilidad con IVA para gestorías .... 90.000,- "
- Contabilidad Analítica .... 50.000,- "
- Base de Datos. Etiquetas. Mailmerge y entronque con Tratamiento de Textos .... 60.000,- "
- Agenda/Caja .... 45.000,- "
- Control de la Gestión de cartera de valores .... 180.000,- "
- Gestión de pisos (alquil. recibos...) .... 110.000,- "
- Concesionarios de coches .... 200.000,- "

Existen muchas Aplicaciones desarrolladas para los PC's a nivel profesional. Si tenéis interés en alguna concreta, llamar al CLUB.

SERVIMOS A TODA ESPAÑA. Mercancía asegurada.

LISTA DE PRECIOS PARA SOCIOS

(Sin 12% IVA)

PRECIOS HARDWARE

ORDENADORES

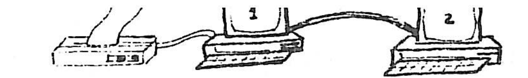
- Oferta SVI-728 (Ordenador, casete, 4 juegos, 1 Joystick)	36.000,- PTAS.
- Operación MEGA (Cambio de cualquier Ordenador por SV-738 y Monitor 12", Disco D/C	82.000,- "
- Compatible SVI-640 FF 640 KB RAM 2 Discos,5 1/4"	169.200,- "
- Compatible SVI-640 FH 640 KB RAM 1 Disco, 5 1/4". Disco Duro 20 MB ...... Agenda electrónica, Base de datos y Procesador de Textos incluído en el precio de los dos SVI	303.300,- "
- Bondwell 32 640 KB RAM 1 Disco de 360 KB. 5 1/4"	120.000,- "
- Bondwell 34 640 KB RAM 2 Discos de 360 KB. 5 1/4"	136.000,- "
- Bondwell 36 640 KB RAM 1 Disco de 360 KB, 5 1/4". Disco Duro de 20 MB	250.000,- "

IMPRESORAS

- Newprint P (paralelo) 80 col. 100 c.p.s.	43.000,- "
- Newprint I (Compatible) 80 col. 130 c.p.s. ...	48.000,- "
- Newprint L (paralelo) 132 col. 100 c.p.s. ...	70.000,- "
- Newprint LI (Compatible) 132 col. 130 c.p.s...	70.000,- "
- C.ITOH 1550 EP (Compatible) 132 col. 120 cps.	115.000,- "
- C.ITOH 1550 SEP(Compatible) 132 col. 180 cps.	120.000,- "
- Brother M-1509. 132 col. 180 c.p.s.	108.000,- "

ACCESORIOS

- Discos de 5 1/4" (Caja de 10)	3.500,- "
- Discos de 3 1/2" (Caja de 10)	5.500,- "
- Filtro de carbono 12"	6.500,- "



REDES DE AREA LOCAL CON PC's

Cuando la pantalla y el teclado de un PC queda desbordado por los tiempos de proceso y la introducción de los datos, existe la posibilidad de conectar otro PC al anterior mediante las correspondientes tarjetas. Las características del que hace de terminal (2) pueden ser mínimas. Por el contrario, el PC maestro (1) ha de tener un Disco Duro para almacenar los programas y la información, accesibles por ambas pantallas y teclados. Se pueden compartir también otros recursos, como por ejemplo la Impresora.

La limitación que tienen este tipo de redes es que cuando se está trabajando con un fichero en una de las pantallas, no puede ser utilizado simultáneamente por la otra. Con lo cual se pierde la posibilidad de consultar al fichero, o realizar la misma Aplicación en las dos pantallas.

Por este motivo, en el caso por ejemplo de una empresa que desee resolver las Aplicaciones de Contabilidad General y la Facturación (Pedidos, Almacén, Albaranes, Facturas), se observa que en la Facturación se utiliza constantemente el fichero del Almacén. La segunda pantalla no resuelve el problema de la saturación. Sería igualmente válido y con un costo inferior, utilizar dos PC's. Cada uno con su Disco Duro.

Incluso con las siguientes ventajas:

- Copiando el fichero del Almacén, ahora sí se puede consultar.
- En caso de avería, no quedaría la solución interrumpida.

Si se trata de un aula informática diseñada para la enseñanza, varios terminales pueden compartir el Disco Duro. La utilización es diferente al caso anteriormente expuesto, porque una vez realizada la función de leer o grabar de un terminal queda liberado el Disco para los demás.

Por último, si los dos PC's se saturan de nuevo, habrá que recurrir a los sistemas multipuesto, diseñados para soportar varios terminales, sin limitaciones a la hora de compartir los ficheros.

Acerca del SORT

En el boletín de Septiembre de 1985 dábamos una breve información sobre el programa SORT de CP/M, mediante un ejemplo de utilización desde MBASIC.

Puesto que el manual de instrucciones de SORT es muy complicado y además está en inglés, consideramos conveniente ampliar aquella información con unas notas que permitan utilizar este programa racionalmente, sin grandes dificultades.

SORT es un potente programa que sirve para ordenar ficheros de CP/M creados con MBASIC, COBOL o cualquier lenguaje o utilidad, así como para mezclar ficheros de las mismas características, reuniéndolos en uno solo, extraer registros con condiciones sobre los campos, crear ficheros de índice, etc.

Actualmente existen versiones de SORT en disco para SVI-318/328 y para SVI-738 (MSX).

El programa SORT funciona de la siguiente manera:

a) Lee el fichero o los ficheros de entrada.

b) Manipula los registros según las instrucciones que se especifiquen, relativas a uno o más campos.

c) Escribe los nuevos registros en el fichero de salida.

Para ello, se deben comunicar a SORT los siguientes datos, como mínimo:

a) Las características o atributos (comunes) de los ficheros de entrada.

b) Los nombres de los ficheros de entrada.

c) El nombre del fichero de salida.

d) Los campos sobre cuyo contenido se dan condiciones de ordenación (campos clave).

Estos datos se suministran mediante ORDENES SORT. Las ORDENES SORT pueden escribirse, bien cada una en una línea, bien varias en la misma línea separadas por punto y coma. La inclusión o no de espacios extra o comas es irrelevante.

A continuación se describen las ORDENES SORT. En lo que sigue, cuando una palabra contenga letras negritas puede abreviarse hasta las negritas solas.

ORDENES PRINCIPALES

1.- INPUT-ATTRIBUTES.

Sirve para especificar las características de los ficheros de entrada.


```
3250 A(X,Y)=V1:IF S1$="E" THEN A(X,Y)=1
3260 GOSUB 1130:GOTO 3100
3270 M$=INKEY$:IF M$="N" THEN GOSUB 3290:GOTO 3270:ELSE IF M$="Z" THEN
     GOSUB 3090:LOCATE 25,20,1:PRINT "                    ";:GOSUB 1120:ELSE RETURN
3280 IF CL$="Y" THEN LOCATE 25,51,0:PRINT "START ";TIME$;:GOTO 340 ELSE GOTO 340
3290 FOR J=1 TO 8:FOR K=1 TO 8:K$=STR$(K)
3300 LOCATE 27-K*3.J*8+1.0
3310 PRINT CHR$(J+64);RIGHT$(K$,1);:NEXT:NEXT:RETURN
3320 GOSUB 3090:LOCATE 25.20,1:PRINT "LEVEL=";B8!;:PRINT "CHANGE LEVELS?";
3330 C$=INKEY$:IF C$="N" THEN GOTO 3370 ELSE IF C$="Y" THEN LOCATE 25,27,1
    :PRINT "         ";:GOSUB 3090:ELSE GOTO 3330
3340 L1$=INKEY$:IF L1$="" THEN GOTO 3340 ELSE IF L1$>="0" AND L1$<="9" THEN
     LOCATE 25,27,1:PRINT L1$;:ELSE GOTO 3340
3350 L2$=INKEY$:IF L2$="" THEN GOTO 3350 ELSE IF L2$<"0" OR L2$>"9" THEN L2$=L1$
    :L1$="0"
3360 B8!=VAL(L1$)*10+VAL(L2$):B7!=B8!/2:LOCATE 25,27,0:PRINT L1$;L2$;
3370 GOSUB 3090:LOCATE 25.20,0:PRINT "                              ";:LOCATE 25,1,1
    :PRINT "MOVE ";B$;:RETURN
3380 REM *** PROMOTION ROUTINE ***
3390 LOCATE 2,1,1:PRINT "P, N, B, R, Q ";
3400 Z$=INKEY$:IF Z$="P" THEN A(A,B)=2:GOTO 3460
3410 IF Z$="N" THEN A(A,B)=4:GOTO 3460
3420 IF Z$="B" THEN A(A,B)=5:GOTO 3460
3430 IF Z$="R" THEN A(A,B)=7:GOTO 3460
3440 IF Z$="Q" THEN A(A,B)=9:GOTO 3460
3450 GOTO 3400
3460 RETURN
3470 IF CL$="Y" THEN LOCATE 25,61,1:PRINT "STOP ";TIME$;
3480 LOCATE 25,1,1:PRINT "HIT (P) TO PLAY OR HIT (B) FOR BASIC";
3490 A$=INKEY$:IF A$="" THEN GOTO 3490 ELSE IF A$="B" THEN CLS:END
     ELSE IF A$="P" THEN RUN ELSE GOTO 3490
3500 CLS
3510 PRINT "1. USE LIST TO INSURE LOADED RIGHT, THE LAST LINE IS 65529 "
3520 PRINT "2. ALSE NOTE MEMORY SIZE AS A DOUBLE CHECK "
3530 PRINT "3. SPECIAL COMMANDS DURING PLAYER'S MOVE ARE:"
3540 PRINT "   TYPE 'N' - TO NUMBER THE BOARD "
3550 PRINT "        'K' - TO CASTLE KING SIDE "
3560 PRINT "        'Q' - TO CASTLE QUEEN SIDE "
3570 PRINT "        'X' - TO EXCHANGE PIECES AND LET COMPUTER MOVE"
3580 PRINT "        'S' - TO LET THE COMPUTER PLAY BY ITSELF "
3590 PRINT "        'M' - TO MODIFY THE BOARD, ENTER THE SQUARE FOLLOWED BY"
3600 PRINT "              C, P, OR E  FOR COMPUTER, PLAYER, OR EMPTY"
3610 PRINT "   .          AND S,P,N,B,R,Q,K  FOR SQUARE,PAWN, ETC..."
3620 PRINT "        'Z' - TO ESCAPE MODIFY AND LET COMPUTER MOVE"
3630 PRINT "        'I' - TO GET INSTRUCTIONS AGAIN"
3640 PRINT "        'L' - TO LOOK AT OR CHANGE LEVEL OF PLAY"
3650 REM ***************************************************************
3660 INPUT "HIT ENTER TO CONTINUE";EN$:PRINT
3670 PRINT "4. TO PROMOTE TO P.N,B.R,Q ENTER THE LETTER WHEN PROMOTED"
3680 PRINT "5. TO CAPTURE 'EN PASSANT' SPECIFY THE 'FROM' - 'TO' SQUARES"
3690 PRINT "6. LEVELS OF PLAY ARE FROM 01 TO 24"
3700 PRINT "7. COMPUTER ASSUMES MATE IF KING IS LEFT IN CHECK SO BE SURE TO "
3710 PRINT "  WATCH FOR CHECK MESSAGE"
3720 PRINT "8. IF A MOVE IS ILLEGAL OR AN ENTRY ERROR IS MADE SIMPLY TYPE IN"
3730 PRINT "   A NEW MOVE.  A MOVE MAY BE STARTED OVER BY FORCING AN ERROR. (E.
G. '9')."
3740 PRINT "9. IF <BREAK> IS HIT DURING THE GAME, TYPE 'GOSUB 900:CONT' TO"
3750 PRINT "   DRAW BOARD AND CONTINUE"
3760 PRINT "10. YOU HAVE A CHOICE OF BLACK OR WHITE, THE COMPUTER WILL ALWAYS"
3770 PRINT "    BE AT THE TOP OF THE SCREEN"
3780 PRINT:PRINT:INPUT "HIT ENTER TO CONTINUE GAME";EN$
3790 RETURN
3800 I=-I:GOSUB 3810:GOSUB 1120:RETURN
3810 L=0:FOR J=1 TO 8:FOR K=1 TO 8:L=L+1:IF A(J,K)<>1 THEN C(L)=-A(J,K)
    :ELSE C(L)=A(J,K)
3820 NEXT:NEXT
3830 L=65:FOR J=1 TO 8: FOR K=1 TO 8:L=L-1:A(J,K)=C(L):NEXT:NEXT
3840 RETURN
3850 END
```

```
2740 PRINT CHR$(32);CHR$(32);CHR$(178);CHR$(32);CHR$(32);CHR$(178);
     CHR$(32);CHR$(32);
2750 LOCATE 26-Y*3,X*8+1,0
2760 PRINT CHR$(32);CHR$(32)+"ROCK"+CHR$(32);CHR$(32);
2770 LOCATE 27-Y*3,X*8+1,0
2780 PRINT STRING$(2,CHR$(32));STRING$(4,CHR$(178));STRING$(2,CHR$(32));
     :RETURN
2790 LOCATE 25-Y*3,X*8+1,0
2800 PRINT CHR$(32);CHR$(32);CHR$(176);CHR$(32);CHR$(32);CHR$(176);
     CHR$(32);CHR$(32);
2810 LOCATE 26-Y*3,X*8+1,0
2820 PRINT CHR$(32);CHR$(32)+"ROOK"+CHR$(32);CHR$(32);
2830 LOCATE 27-Y*3,X*8+1,0
2840 PRINT STRING$(2,CHR$(32));STRING$(4,CHR$(176));STRING$(2,CHR$(32));
     :RETURN
2850 LOCATE 25,31,1:PRINT STRING$(10," ");
2860 X$=INKEY$:IF X$="N" THEN N$=X$
2870 IF X$>="A" AND X$<="H" THEN X=VAL(CHR$(ASC(X$)-16)):GOTO 2990
2880 IF X$="N" THEN GOSUB 3290
2890 IF X$="K" AND A(5,1)=99 AND A(8,1)=7 THEN A(5,1)=1:A(8,1)=1:A(7,1)=99
     :A(6,1)=7:GOTO 3060
2900 IF X$="Q" AND A(5,1)=99 AND A(1,1)=7 THEN A(5,1)=1:A(1,1)=1:A(3,1)=99
     :A(4,1)=7:GOTO 3060
2910 IF X$="K" AND A(4,1)=99 AND A(1,1)=7 THEN A(4,1)=1:A(1,1)=1:A(2,1)=99
     :A(3,1)=7:GOTO 3060
2920 IF X$="Q" AND A(4,1)=99 AND A(8,1)=7 THEN A(4,1)=1:A(8,1)=1:A(6,1)=99
     :A(5,1)=7:GOTO 3060
2930 IF X$="X" THEN GOSUB 3090:LOCATE 25,1,0:PRINT "EXCHANGING          ";
     :GOSUB 3800:IF CL$="Y" THEN LOCATE 25,50,0:PRINT "START ";TIME$;:GOTO 340
     :ELSE:GOTO 340
2940 IF X$="S" THEN GOTO 3080
2950 IF X$="M" THEN GOSUB 3100
2960 IF X$="I" THEN GOSUB 3500:CLS:GOSUB 1120:LOCATE 25,1,1
     :PRINT "MOVE "+CHR$(7);B$;
2970 IF X$="L" THEN GOSUB 3320
2980 GOTO 2860
2990 LOCATE 25,21,1:PRINT X$;:LOCATE 25,31,1:PRINT "          ";
3000 Y=VAL(INKEY$):IF Y=0 THEN GOTO 3000
3010 LOCATE 25,22,1:PRINT Y;:LOCATE 25,25,1:PRINT "-";
3020 IF A(X,Y)<2 OR A(X,Y)>99 THEN LOCATE 25,31,1:PRINT "ENTRY ERROR ";
     :GOTO 2850
3030 A$=INKEY$:IF A$<"A" OR A$>"H" THEN GOTO 3030 ELSE A=VAL(CHR$(ASC(A$)-16))
3040 LOCATE 25,27,1:PRINT A$;
3050 B=VAL(INKEY$):IF B=0 THEN GOTO 3050 ELSE LOCATE 25,28,1:PRINT B;
3060 IF X$="K" OR X$="Q" THEN LOCATE 25,20,1:PRINT "CSTL. ";X$;"-SIDE";
     :GOSUB 3090 ELSE GOSUB 3090
3070 IF CL$="Y" THEN LOCATE 25,50,0:PRINT "START ";TIME$;
3080 RETURN
3090 LOCATE 25,1,1:PRINT STRING$(79," ");:RETURN
3100 GOSUB 3090:LOCATE 25,1,1:PRINT "MODIFYING          ";
3110 GOSUB 3270:IF M$<"A" OR M$>"H" THEN GOTO 3110
3120 X=VAL(CHR$(ASC(M$)-16)):LOCATE 25,21,1:PRINT M$:
3130 M$=INKEY$:IF M$<"1" OR M$>"8" THEN GOTO 3130
3140 Y=VAL(M$):LOCATE 25,23,1:PRINT M$;" = ";
3150 M$=INKEY$:IF M$="C" OR M$="P" OR M$="E" THEN S1$=M$:LOCATE 25,27,1
     :PRINT M$;:ELSE GOTO 3150
3160 M$=INKEY$:IF M$="S" OR M$="P" OR M$="N" OR M$="B" OR M$="R" OR M$="Q" OR
     M$="K" THEN V$=M$:LOCATE 25,29,1:PRINT M$; ELSE GOTO 3160
3170 IF V$="P" THEN V1=2
3180 IF V$="N" THEN V1=4
3190 IF V$="B" THEN V1=5
3200 IF V$="R" THEN V1=7
3210 IF V$="Q" THEN V1=9
3220 IF V$="K" THEN V1=99
3230 IF V$="S" THEN V1=1
3240 IF S1$="C" AND V1<>1 THEN V1=-V1
```


Forma de la orden:
INPUT-ATTRIBUTES=long tipo

Observaciones:

a) El signo = puede sustituirse por un espacio.

b) <u>long</u> es un parámetro obligatorio de la orden. Debe ser un numero entre 1 y 4096 y especifica la longitud de los registros. (En ficheros secuenciales especifica la longitud máxima).

c) <u>tipo</u> es un parámetro opcional, que puede ser uno de los siguientes:

FIXED-LENGTH	-fichero de registros de longitud fija, por ejemplo: los ficheros de acceso directo de MBASIC. Se supone por defecto.
CR-DELIMITED	-fichero de registros separados por ODH y/o OAH, y campos separados por comas, como los ficheros secuenciales de MBASIC con la orden WRITE#.
VARIABLE	-ficheros secuenciales COBOL de longitud variable.
RELATIVE	-ficheros relativos COBOL.

Ejemplos:

INP 63, CR-DEL
INPUT-ATTR=104
I 42 FI

2.- <u>SORT-FILES.</u>

Sirve para especificar los nombres de los ficheros de entrada que deben ordenarse.

Forma:

SORT-FILES=fich1(com fin) fich2(com fin) ...

Observaciones:

a) Cada orden SORT-FILES seguida de un = o un espacio anula cualquier orden SORT-FILES anterior, pero si va seguida del signo + sirve para agregar ficheros de entrada a los anteriores.

b) <u>fich1</u>, <u>fich2</u>, ... son los nombres de los ficheros de entrada. SORT admite hasta 32 ficheros de entrada.

c) <u>com</u>, <u>fin</u> son dos números opcionales entre 1 y 65535 que indican el comienzo y el final del rango de registros que deben ser procesados y anotados en el fichero de salida.

Ejemplos:

SORT A:CLIENTES.DAT
S=CLIENTES.DAT(3,860),B:ALTAS.DAT
S+B:INCIDEN.DAT

3.- <u>MERGE-FILES.</u>

Sirve para especificar los nombres de los ficheros de entrada que no deben ordenarse (por encon-

trarse ordenados previamente), y deben combinarse para producir el fichero de salida ordenado.

Forma:

MERGE-FILES=fich1 fich2 ...

Observaciones:

a) Cada orden MERGE-FILES seguida de un = o un espacio anula cualquier orden MERGE-FILES anterior, pero si va seguida del signo + sirve para agregar ficheros de entrada a los anteriores.

b) fich1, fich2, ... son nombres de los ficheros de entrada.

c) La orden MERGE-FILES no admite parametros de rango de registros.

Ejemplos:

MERG A:CLIENTES.DAT, B:ALTAS.DAT
M=CLIENTES.DAT,B:ALTAS.1
M+ALTAS2

4.- KEY.

Sirve para especificar los campos clave y las condiciones de ordenación.

Forma:

KEY=campo1 cond1 campo2 cond2 ...

Observaciones:

a) Cada orden KEY seguida de un = o un espacio anula cualquier orden KEY anterior, pero si va seguida del signo + sirve para agregar campos clave a los anteriores. Los campos clave deben aparecer por orden de prioridad. Pueden especificarse hasta 32 campos clave.

b) campo1, campo2, etc. tienen la siguiente forma:

#num long	-número de orden del campo en el registro y longitud máxima del campo (para ficheros de tipo CR-DELIMITED).
com fin	-números de posición primera y última del campo en el registro.

c) cond1, cond2, etc. son condiciones opcionales de ordenación entre las siguientes posibles:

ASCENDING	-Orden natural. Se supone por defecto.
DESCENDING	-Orden inverso del anterior.
NUMERIC-ASCII	-El campo es numérico.
UPPER-CASE	-Considera las minúsculas como mayúsculas para ordenación.
RIGHT-JUSTIFY	-Campo alineado por la derecha.
LOHI	-Ordena de derecha a izquierda (no válido para CR-DELIMITED).
MASK-PARITY-BIT	-Ignora el 1er. bit de cada byte.


```
2190 PRINT STRING$(3,CHR$(219));STRING$(2,CHR$(176));STRING$(3,CHR$(219));
     :RETURN
2200 IF I*A(X,Y)<0 THEN GOTO 2270
2210 LOCATE 25-Y*3,X*8+1,0
2220 PRINT STRING$(3,CHR$(32));STRING$(2,CHR$(178));STRING$(3,CHR$(32));
2230 LOCATE 26-Y*3,X*8+1,0
2240 PRINT CHR$(32)+"KNIGHT"+CHR$(32);
2250 LOCATE 27-Y*3,X*8+1,0
2260 PRINT STRING$(3,CHR$(32));STRING$(2,CHR$(178));STRING$(3,CHR$(32));
     :RETURN
2270 LOCATE 25-Y*3,X*8+1,0
2280 PRINT STRING$(3,CHR$(32));STRING$(2,CHR$(176));STRING$(3,CHR$(32));
2290 LOCATE 26-Y*3,X*8+1,0
2300 PRINT CHR$(32)+"KNIGHT"+CHR$(32);
2310 LOCATE 27-Y*3,X*8+1,0
2320 PRINT STRING$(3,CHR$(32));STRING$(2,CHR$(176));STRING$(3,CHR$(32));
     :RETURN
2330 IF I*A(X,Y)<0 THEN GOTO 2400
2340 LOCATE 25-Y*3,X*8+1,0
2350 PRINT STRING$(3,CHR$(219));STRING$(2,CHR$(178));STRING$(3,CHR$(219));
2360 LOCATE 26-Y*3,X*8+1,0
2370 PRINT CHR$(219)+"BISHOP"+CHR$(219);
2380 LOCATE 27-Y*3,X*8+1,0
2390 PRINT STRING$(3,CHR$(219));STRING$(2,CHR$(178));STRING$(3,CHR$(219));
     :RETURN
2400 LOCATE 25-Y*3,X*8+1,0
2410 PRINT STRING$(3,CHR$(219));STRING$(2,CHR$(176));STRING$(3,CHR$(219));
2420 LOCATE 26-Y*3,X*8+1,0
2430 PRINT CHR$(219)+"BISHOP"+CHR$(219);
2440 LOCATE 27-Y*3,X*8+1,0
2450 PRINT STRING$(3,CHR$(219));STRING$(2,CHR$(176));STRING$(3,CHR$(219));
     :RETURN
2460 IF I*A(X,Y)<0 THEN GOTO 2530
2470 LOCATE 25-Y*3,X*8+1,0
2480 PRINT STRING$(3,CHR$(32));STRING$(2,CHR$(178));STRING$(3,CHR$(32));
2490 LOCATE 26-Y*3,X*8+1,0
2500 PRINT CHR$(32)+"BISHOP"+CHR$(32);
2510 LOCATE 27-Y*3,X*8+1,0
2520 PRINT STRING$(3,CHR$(32));STRING$(2,CHR$(178));STRING$(3,CHR$(32));
     :RETURN
2530 LOCATE 25-Y*3,X*8+1,0
2540 PRINT STRING$(3,CHR$(32));STRING$(2,CHR$(176));STRING$(3,CHR$(32));
2550 LOCATE 26-Y*3,X*8+1,0
2560 PRINT CHR$(32)+"BISHOP"+CHR$(32);
2570 LOCATE 27-Y*3,X*8+1,0
2580 PRINT STRING$(3,CHR$(32));STRING$(2,CHR$(176));STRING$(3,CHR$(32));
     :RETURN
2590 IF I*A(X,Y)<0 THEN GOTO 2660
2600 LOCATE 25-Y*3,X*8+1,0
2610 PRINT CHR$(219);CHR$(219);CHR$(178);CHR$(219);CHR$(219);CHR$(178);
     CHR$(219);CHR$(219);
2620 LOCATE 26-Y*3,X*8+1,0
2630 PRINT CHR$(219);CHR$(219)+"ROOK"+CHR$(219);CHR$(219);
2640 LOCATE 27-Y*3,X*8+1,0
2650 PRINT STRING$(2,CHR$(219));STRING$(4,CHR$(178));STRING$(2,CHR$(219));
     :RETURN
2660 LOCATE 25-Y*3,X*8+1,0
2670 PRINT CHR$(219);CHR$(219);CHR$(176);CHR$(219);CHR$(219);CHR$(176);
     CHR$(219);CHR$(219);
2680 LOCATE 26-Y*3,X*8+1,0
2690 PRINT CHR$(219);CHR$(219)+"ROOK"+CHR$(219);CHR$(219);
2700 LOCATE 27-Y*3,X*8+1,0
2710 PRINT STRING$(2,CHR$(219));STRING$(4,CHR$(176));STRING$(2,CHR$(219));
     :RETURN
2720 IF I*A(X,Y)<0 THEN GOTO 2790
2730 LOCATE 25-Y*3,X*8+1,0
```

```
      :RETURN
1620 LOCATE 25-Y*3,X*8+1,0
1630 PRINT STRING$(2,CHR$(219));STRING$(3,CHR$(176));STRING$(3,CHR$(219));
1640 LOCATE 26-Y*3,X*8+1,0
1650 PRINT CHR$(219)+"QUEEN"+CHR$(219);CHR$(219);
1660 LOCATE 27-Y*3,X*8+1,0
1670 PRINT STRING$(2,CHR$(219));STRING$(3,CHR$(176));STRING$(3,CHR$(219));
      :RETURN
1680 IF I*A(X,Y)<0 THEN GOTO 1750
1690 LOCATE 25-Y*3,X*8+1,0
1700 PRINT STRING$(2,CHR$(32));STRING$(3,CHR$(178));STRING$(3,CHR$(32));
1710 LOCATE 26-Y*3,X*8+1,0
1720 PRINT CHR$(32)+"QUEEN"+CHR$(32);CHR$(32);
1730 LOCATE 27-Y*3,X*8+1,0
1740 PRINT STRING$(2,CHR$(32));STRING$(3,CHR$(178));STRING$(3,CHR$(32));
      :RETURN
1750 LOCATE 25-Y*3,X*8+1,0
1760 PRINT STRING$(2,CHR$(32));STRING$(3,CHR$(176));STRING$(3,CHR$(32));
1770 LOCATE 26-Y*3,X*8+1,0
1780 PRINT CHR$(32)+"QUEEN"+CHR$(32);CHR$(32);
1790 LOCATE 27-Y*3,X*8+1,0
1800 PRINT STRING$(2,CHR$(32));STRING$(3,CHR$(176));STRING$(3,CHR$(32));
      :RETURN
1810 IF I*A(X,Y)<0 THEN GOTO 1880
1820 LOCATE 25-Y*3,X*8+1,0
1830 PRINT STRING$(2,CHR$(219));STRING$(4,CHR$(178));STRING$(2,CHR$(219));
1840 LOCATE 26-Y*3,X*8+1,0
1850 PRINT CHR$(219);CHR$(178)+"KING"+CHR$(178);CHR$(219);
1860 LOCATE 27-Y*3,X*8+1,0
1870 PRINT STRING$(2,CHR$(219));STRING$(4,CHR$(178));STRING$(2,CHR$(219));
      :RETURN
1880 LOCATE 25-Y*3,X*8+1,0
1890 PRINT STRING$(2,CHR$(219));STRING$(4,CHR$(176));STRING$(2,CHR$(219));
1900 LOCATE 26-Y*3,X*8+1,0
1910 PRINT CHR$(219);CHR$(176)+"KING"+CHR$(176);CHR$(219);
1920 LOCATE 27-Y*3,X*8+1,0
1930 PRINT STRING$(2,CHR$(219));STRING$(4,CHR$(176));STRING$(2,CHR$(219));
      :RETURN
1940 IF I*A(X,Y)<0 THEN GOTO 2010
1950 LOCATE 25-Y*3,X*8+1,0
1960 PRINT STRING$(2,CHR$(32));STRING$(4,CHR$(178));STRING$(2,CHR$(32));
1970 LOCATE 26-Y*3,X*8+1,0
1980 PRINT CHR$(32);CHR$(178)+"KING"+CHR$(178);CHR$(32);
1990 LOCATE 27-Y*3,X*8+1,0
2000 PRINT STRING$(2,CHR$(32));STRING$(4,CHR$(178));STRING$(2,CHR$(32));
      :RETURN
2010 LOCATE 25-Y*3,X*8+1,0
2020 PRINT STRING$(2,CHR$(32));STRING$(4,CHR$(176));STRING$(2,CHR$(32));
2030 LOCATE 26-Y*3,X*8+1,0
2040 PRINT CHR$(32);CHR$(176)+"KING"+CHR$(176);CHR$(32);
2050 LOCATE 27-Y*3,X*8+1,0
2060 PRINT STRING$(2,CHR$(32));STRING$(4,CHR$(176));STRING$(2,CHR$(32));
      :RETURN
2070 IF I*A(X,Y)<0 THEN GOTO 2140
2080 LOCATE 25-Y*3,X*8+1,0
2090 PRINT STRING$(3,CHR$(219));STRING$(2,CHR$(178));STRING$(3,CHR$(219));
2100 LOCATE 26-Y*3,X*8+1,0
2110 PRINT CHR$(219)+"KNIGHT"+CHR$(219);
2120 LOCATE 27-Y*3,X*8+1,0
2130 PRINT STRING$(3,CHR$(219));STRING$(2,CHR$(178));STRING$(3,CHR$(219));
      :RETURN
2140 LOCATE 25-Y*3,X*8+1,0
2150 PRINT STRING$(3,CHR$(219));STRING$(2,CHR$(176));STRING$(3,CHR$(219));
2160 LOCATE 26-Y*3,X*8+1,0
2170 PRINT CHR$(219)+"KNIGHT"+CHR$(219);
2180 LOCATE 27-Y*3,X*8+1,0
```


EBCDIC	-Ordena con arreglo al código EBCDIC.
ALTSEQ	-Ordena con arreglo a una sucesión definida por el usuario.
COMPUTATIONAL	-Campo numérico COBOL de 2 bytes.
INTEGER	-Campo entero de MBASIC o FORTRAN de 2 bytes.
FLOATING-POINT	-Campo real, simple precisión o doble precisión de 4 u 8 bytes MBASIC o FORTRAN.
COMPUTATIONAL-3	-Campo numérico COBOL de tipo BCD.

Ejemplos:

```
KEY=2 10 ASC, 20 45 DESC
KEY #11,40,ASC #3,5,DESC #6,8,ASC
K + #2,10 DE
K #17,10 NUM,DESC #8,20 ASCEND,RIGHT,UPPER
```

5.- <u>OUTPUT-FILE.</u>

Sirve para especificar el nombre (y opcionalmente los atributos) del fichero de salida.

Forma:

OUTPUT-FILE=fichero parámetros

Observaciones:

a) El signo = puede sustituirse por un espacio.

b) <u>fichero</u> es el nombre del fichero de salida. Si va seguido de /C permite cambiar el disco para recibirlo, cuando aparece el mensaje apropiado.

c) Los parámetros son todos opcionales, y pueden ser, en este orden:

- un número que especifica la longitud del fichero nuevo
- el tipo del fichero nuevo (FI, CR, ...)
- Los campos que se seleccionan para el fichero nuevo, con arreglo a las siguientes posibilidades:

K-OUTPUT	-sólo los campos clave
R-OUTPUT	-sólo números de registro
P-OUTPUT	-sólo punteros: (num-sector, posición)
KR-OUTPUT	-sólo claves y números de registro
KP-OUTPUT	-sólo claves y punteros.

d) Si se omite alguno de los parámetros o todos ellos, su valor por defecto es el mismo de los ficheros de entrada.

Ejemplos:

```
OUTP=CLIENTES.ACT
O=B:NUEVO.FCH/C,200
O CLAVE1 15 CR KR
```

6.- <u>GO.</u>

Sirve para comenzar la ejecución con arreglo a

los datos suministrados mediante las demás órdenes. Debe ser, por tanto, la orden final.

EJEMPLOS DE SESIONES SORT

```
A>SORT
* INPUT-ATTR = 62, CR-DEL
* SORT-FILE = ALMACEN
* KEY = #3,4
* OUTPUT-FILE = ALMACEN
* GO
------
A>SORT
* INP=80, FIX
* SORT ALUMNOS.NEW
* MERGE ALUMNOS.DAT
* KEY=72,80 DES, 50,71 ASC, 1,49 ASC,UPPER
* OUTPUT ALUMNOS.DAT
* GO
------
A>SORT I 60 FI;S B:PAGOS;K 1 5, 9 20, 8 8;O B:PAGOS;G
```

OTRAS ORDENES SORT

CFILE=fichero
Introduce las ORDENES SORT contenidas en un fichero de texto. Si se quiere que éstas aparezcan en pantalla, debe especificarse previamente con la orden:
LIST

WORK-DRIVE=disco
Permite que el fichero temporal de trabajo SORT.$$$ se almacene en la unidad de disco especificada. Por defecto es el disco en curso. Si se utiliza la opcion /C debe ser distinto al del fichero de salida.

TAGSORT
Para procesos con sólo un fichero SORT-FILES y ningún fichero MERGE-FILES, reduce el espacio del fichero de trabajo, sin embargo es más lento.

PRINT-LEVEL=n
Cambia la amplitud de los mensajes que aparecen en el curso de una sesión SORT. Por defecto n=2, pudiendo valer de 0 a 5.

NO-ERROR-MESSAGES
No presenta mensajes de error, ganando espacio para la zona de trabajo y mejorando la velocidad.

RETURN-TO-CONSOLE
Permite encadenar sesiones SORT sin salir al sistema operativo.

COLLATING-SEQUENCE=argumentos
Sirve para definir una sucesión alternativa de caracteres para ordenación con la condición ALTSEQ de la orden KEY.
Si los argumentos van entre comillas, se refie-


```
     T(X,Y)=K:NEXT:NEXT
1030 Y=0:FOR J=960 TO 64 STEP -128:Y=Y+1:X=0:FOR K=J TO 42+J STEP 6:X=X+1:
      B(X,Y)=K:NEXT:NEXT
1040 DATA -7, -4, -5, -9, -99, -5, -4, -7
1050 DATA -2, -2, -2, -2, -2, -2, -2, -2
1060 DATA 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1
1070 DATA 2, 2, 2, 2, 2, 2, 2, 2
1080 DATA 7, 4, 5, 9, 99, 5, 4, 7
1090 FOR Y=8 TO 1 STEP -1:FOR X=1 TO 7 STEP 2:S(X,Y)=I:S(X+1,Y)=-I:NEXT:I=-I:NEX
T
1100 FOR Y=8 TO 1 STEP -1:FOR X=1 TO 8:READ A(X,Y):NEXT:NEXT:RETURN
1110 I=-I:A(4,1)=99:A(5,1)=9:A(4,8)=-99:A(5,8)=-9
1120 FOR Y=1 TO 8:FOR X=1 TO 8:GOSUB 1130:NEXT:NEXT:RETURN
1130 IF S(X,Y)>0 THEN GOTO 1150
1140 ON ABS(A(X,Y)) GOSUB 1230, 1420, 0, 2200, 2460, 0, 2720, 0, 1680
        :IF ABS(A(X,Y))=99 THEN GOSUB 1940:RETURN ELSE RETURN
1150 ON ABS(A(X,Y)) GOSUB 1160, 1290, 0, 2070, 2330, 0, 2590, 0, 1550
         :IF ABS(A(X,Y))=99 THEN GOSUB 1810:RETURN ELSE RETURN
1160 IF S(X,Y)<0 THEN GOTO 1230
1170 LOCATE 25-Y*3,X*8+1,0
1180 PRINT STRING$(8,CHR$(219));
1190 LOCATE 26-Y*3,X*8+1,0
1200 PRINT STRING$(8,CHR$(219));
1210 LOCATE 27-Y*3,X*8+1,0
1220 PRINT STRING$(8,CHR$(219));:RETURN
1230 LOCATE 25-Y*3,X*8+1,0
1240 PRINT STRING$(8," ");
1250 LOCATE 26-Y*3,X*8+1,0
1260 PRINT STRING$(8," ");
1270 LOCATE 27-Y*3,X*8+1,0
1280 PRINT STRING$(8," ");:RETURN
1290 IF I*A(X,Y)<0 THEN GOTO 1360
1300 LOCATE 25-Y*3,X*8+1,0
1310 PRINT STRING$(8,CHR$(219));
1320 LOCATE 26-Y*3,X*8+1,0
1330 PRINT STRING$(3,CHR$(219));CHR$(2);STRING$(4,CHR$(219));
1340 LOCATE 27-Y*3,X*8+1,0
1350 PRINT STRING$(8,CHR$(219)):RETURN
1360 LOCATE 25-Y*3,X*8+1,0
1370 PRINT STRING$(8,CHR$(219));
1380 LOCATE 26-Y*3,X*8+1,0
1390 PRINT STRING$(3,CHR$(219));CHR$(1);STRING$(4,CHR$(219));
1400 LOCATE 27-Y*3,X*8+1,0
1410 PRINT STRING$(8,CHR$(219));:RETURN
1420 IF I*A(X,Y)<0 THEN GOTO 1490
1430 LOCATE 25-Y*3,X*8+1,0
1440 PRINT STRING$(8,CHR$(32));
1450 LOCATE 26-Y*3,X*8+1,0
1460 PRINT STRING$(3,CHR$(32));CHR$(2);STRING$(4,CHR$(32));
1470 LOCATE 27-Y*3,X*8+1,0
1480 PRINT STRING$(8,CHR$(32)):RETURN
1490 LOCATE 25-Y*3,X*8+1,0
1500 PRINT STRING$(8,CHR$(32));
1510 LOCATE 26-Y*3,X*8+1,0
1520 PRINT STRING$(3,CHR$(32));CHR$(1);STRING$(4,CHR$(32));
1530 LOCATE 27-Y*3,X*8+1,0
1540 PRINT STRING$(8,CHR$(32));:RETURN
1550 IF I*A(X,Y)<0 THEN GOTO 1620
1560 LOCATE 25-Y*3,X*8+1,0
1570 PRINT STRING$(2,CHR$(219));STRING$(3,CHR$(178));STRING$(3,CHR$(219));
1580 LOCATE 26-Y*3,X*8+1,0
1590 PRINT CHR$(219)+"QUEEN"+CHR$(219);CHR$(219);
1600 LOCATE 27-Y*3,X*8+1,0
1610 PRINT STRING$(2,CHR$(219));STRING$(3,CHR$(178));STRING$(3,CHR$(219));
```

```
470 IF A(A,B)=2 AND B=8 THEN LOCATE 25,31,1:PRINT "WHAT PIECE";:GOSUB 3380:
    X=A:Y=B:GOSUB 1130:LOCATE 25,31,1:PRINT "              ";
480 IF B5=1 THEN A(A,B-1)=1:X=A:Y=B-1:GOSUB 1130:GOTO 340:ELSE GOTO 340
490 LOCATE 1,78,0:PRINT "$ ";:
    ON A(X,Y) GOTO 0, 700, 0, 600, 550, 0, 510, 0, 550:GOTO 690
500 ON -A(X,Y) GOTO 0, 750, 0, 600, 550, 0, 510, 0, 550:GOTO 680
510 B=Y:FOR A=X+1 TO 8:GOSUB 860:IF S=0 THEN NEXT ELSE GOTO 520
520 FOR A=X-1 TO 1 STEP -1:GOSUB 860:IF S=0 THEN NEXT ELSE GOTO 530
530 A=X:FOR B=Y+1 TO 8:GOSUB 860:IF S=0 THEN NEXT ELSE GOTO 540
540 FOR B=Y-1 TO 1 STEP -1:GOSUB 860:IF S=0 THEN NEXT:RETURN ELSE RETURN
550 B=Y:FOR A=X+1 TO 8:B=B+1:GOSUB 860:IF S=0 THEN NEXT ELSE GOTO 560
560 B=Y:FOR A=X-1 TO 1 STEP -1:B=B-1:GOSUB 860:IF S=0 THEN NEXT ELSE GOTO 570
570 B=Y:FOR A=X-1 TO 1 STEP -1:B=B+1:GOSUB 860:IF S=0 THEN NEXT ELSE GOTO 580
580 B=Y:FOR A=X+1 TO 8:B=B-1:GOSUB 860: IF S=0 THEN NEXT ELSE GOTO 590
590 IF ABS(A(X,Y))=9 THEN GOTO 510:ELSE RETURN
600 A=X+2:B=Y+1:IF A<9 AND B<9 THEN GOSUB 870 ELSE GOTO 610
610 B=B-2:IF B>0 AND A<9 THEN GOSUB 870 ELSE GOTO 620
620 A=A-4:IF A>0 AND B>0 THEN GOSUB 870 ELSE GOTO 630
630 B=B+2:IF B<9 AND A>0 THEN GOSUB 870 ELSE GOTO 640
640 A=A+1:B=B+1:IF A>0 AND A<9 AND B<9 THEN GOSUB 870 ELSE GOTO 650
650 B=B-4:IF B>0 AND A>0 AND A<9 THEN GOSUB 870 ELSE GOTO 660
660 A=A+2:IF A>0 AND A<9 AND B>0 THEN GOSUB 870 ELSE GOTO 670
670 B=B+4:IF B<9 AND A>0 AND A<9 THEN GOSUB 870:RETURN ELSE RETURN
680 FOR A=X-1 TO X+1:FOR B=Y-1 TO Y+1:IF A(A,B)<>0 THEN GOSUB 870
690 NEXT:NEXT:RETURN
700 A=X:IF Y>2 THEN GOTO 720 ELSE GOTO 710
710 B=Y+1:IF A(A,B)=1 THEN GOSUB 880:B=B+1:IF A(A,B)=1 THEN GOSUB 880:GOTO 730:
    ELSE GOTO 730:ELSE GOTO 730
720 B=Y+1:IF A(A,B)=1 THEN GOSUB 880 ELSE GOTO 730
730 A=X+1:B=Y+1:IF A(A,B)<0 THEN GOSUB 880 ELSE GOTO 740
740 A=A-2:IF A(A,B)<0 THEN GOSUB 880:RETURN:ELSE RETURN
750 A=X:IF Y<7 THEN GOTO 770 ELSE GOTO 760
760 B=Y-1:IF A(A,B)=1 THEN GOSUB 860:B=B-1:IF A(A,B)=1 THEN GOSUB 880:GOTO 780:
    ELSE GOTO 780:ELSE GOTO 790
770 B=Y-1:IF A(A,B)=1 THEN GOSUB 880 ELSE GOTO 780
780 A=X-1:B=Y-1:IF A(A,B)>1 THEN GOSUB 880 ELSE GOTO 790
790 A=A+2:IF A(A,B)>1 THEN GOSUB 880:RETURN:ELSE GOTO 810
800 T=A(A,B):IF T=-99 THEN B1=T:RETURN:ELSE GOTO 810
810 A5=S:IF ABS(T)<=A(X,Y) THEN A(A,B)=A(X,Y):A(X,Y)=1:GOTO 830:ELSE GOTO 820
820 IF T<B1 THEN B1=T:S=A5:RETURN:ELSE S=A5:RETURN
830 A1=X:A2=Y:A3=A:A4=B:A8=T:A0=2:FOR X=1 TO 8:FOR Y=1 TO 8:
    IF A(X,Y)<0 THEN GOSUB 500:IF T=0 THEN GOTO 850
840 NEXT:NEXT
850 X=A1:Y=A2:A=A3:B=A4:A0=5:A(X,Y)=A(A,B):A(A,B)=A8:GOTO 820
860 S=0:IF A(A,B)=1 THEN GOTO 880 ELSE IF A(A,B)=0 THEN S=1:RETURN:
    ELSE IF SGN(A(A,B))=SGN(A(X,Y)) THEN S=1:RETURN:ELSE S=1:GOTO 880
870 IF A(A,B)=1 THEN GOTO 880 ELSE IF SGN(A(A,B))=SGN(A(X,Y)) THEN RETURN:
    ELSE GOTO 880
880 LOCATE 1,78,0:PRINT "  ";:ON A0 GOTO 890, 900, 910, 920, 930:GOTO 940
890 IF A6=A AND A7=B THEN B1=B1+1:RETURN:ELSE RETURN
900 IF A3=A AND A4=B THEN T=0:RETURN:ELSE RETURN
910 IF B4=A AND B9=B THEN D=1:RETURN:ELSE RETURN
920 IF A6=A AND A7=B THEN C=1:RETURN:ELSE RETURN
930 IF A(A,B)<0 THEN GOTO 800 ELSE RETURN
940 B3=S:W=X:M=Y:N=A:H=B:P=A(A,B):A(A,B)=A(X,Y):A(X,Y)=1:B1=0
950 A0=5:FOR X=1 TO 8:FOR Y=1 TO 8:IF A(X,Y)>1 THEN GOSUB 490
960 NEXT:NEXT:X=N:Y=H:A0=1:GOSUB 500:A0=0:S=B3:X=W:Y=M:A=N:B=H:A(X,Y)=A(A,B):
    A(A,B)=P
970 B6!=1/(ABS(4.5-A)+ABS(4.5-B)+1):IF A(X,Y)<-2 AND A(X,Y)>-9 THEN
    B6!=B6!+1/(ABS(A6-A)+ABS(A7-B)+5)+RND(0)/15
980 G!=P+B1+B6!:IF P=99 THEN GOSUB 1120:
    LOCATE 25,1,1:PRINT "MATE ";B$;:GOTO 3470
990 IF G!<=F! THEN RETURN
1000 F!=G!:E=X:Q=Y:R=A:U=B:RETURN
1010 CLS
1020 Y=0:FOR J=896 TO 0 STEP -128:Y=Y+1:X=0:FOR K=J TO 42+J STEP 6:X=X+1:
```


ren al carácter ASCII que contienen. En otro caso son los números de los caracteres a que se refieren.

Ejemplo:

COL=0-47,58-64,91-95,127-256,"a"-"z","A"-"Z","9"-"0"

tiene primero los caracteres especiales y de puntuación, después las minúsculas seguidas de las mayúsculas y finalmente los números con el orden invertido.

SELECT=condiciones

Sirve para seleccionar los registros que verifiquen condiciones relativas a uno o varios de sus campos.

Observaciones:

a) Cada orden SELECT seguida de un = o un espacio anula cualquier orden SELECT anterior, pero si va seguida del signo + sirve para agregar condiciones a las anteriores.

b) Las condiciones se forman igual que en MBASIC, con los símbolos <, =, >, <=, >=, <>, los operadores AND, OR, XOR, NOT, los paréntesis y además los operadores BT (comprendido entre) y NB (no comprendido entre).

c) Los campos sujetos a condiciones se especifican mediante:

#num	-para los ficheros que sean de tipo CR-DELIMITED.
FIELD com fin	-para los demás tipos de ficheros.

Ejemplos:

```
SELECT FIELD 1 18 < "G" AND FIELD 19 25 > "3000" NUM
SE #1 BT "12000","50000" NUM-ASC
SE (#1 >= #2 OR #2 BT #3,#4) AND #6 = "Viudo" UPPER
SE FI 16 16 = 13       (byte 13 = ^M, no número "13").
```

EXCLUDE=condiciones

Sirve para excluir los registros que verifiquen condiciones relativas a uno o varios de sus campos.

A la orden EXCLUDE se le aplican las mismas observaciones que a SELECT.

Importante:

Sobre ficheros de acceso directo MBASIC.

Un fichero de acceso directo no tiene marca de final. Desde el último registro, el sector del disco se llena con el byte 0. Si tal fichero se ordena, pueden pasar varios registros ficticios llenos de bytes 0 a la cabecera. Esto puede evitarse mediante una orden de selección, por ejemplo:

```
A>SORT
* INPUT = 140 FIXED
* SELECT FIELD 1 140 > 0
*   ...
```

RUTINAS DEL Z-80

Agradecemos a Juan Carlos Amorós de Barcelona estas rutinas, fruto de su magnífico esfuerzo que puede ser interesante para todos los amigos del Club I.S.I.S.

Estas son algunas de las utilidades en codigo maquina que se pueden utilizar en la confeccion de programas:

```
*-Para  leer  del  teclado:
        Call 403D        ;Obtenemos el caracter ASCII de
                          la tecla pulsada, en el regis-
                          tro A

*-Para escribir en pantalla:
        Rst  18          ;Introduciendo en el registro A
                          el caracter  ASCII,  antes  de
                          acceder a la llamada Rst 18.
                          (No altera ningun registro)

*-Para escribir en impresora:
        Ld A,01          ;00 pantalla, 01 impresora
        Ld(F542),A       ;PRTFLG Printer flag (Variable)

        Rst 18           ;Igual que antes pero en impre-
                          sora,  mientras  se  mantenga
                          PRTFLG a 01.

*-Para detener programa con un mensaje de error:
        Jp 03ED          ;Syntax error
        Jp 08F0          ;Division by zero
        Jp 08F6          ;Redimensioned array
        Jp 08FF          ;Overflow
        Jp 0902          ;Missing operand
        Jp 0905          ;Type mismatch

*-Para detener programa con codigo de error:
        Ld,E             ;En el registro  E debe figurar
                          el numero de codigo de error.
        Jp 0907          ;Llamada rutina detencion error.

*-Para leer del teclado ,teclas especiales:
        Ld A,(FD86)      ;SFTKEY Shift key (Variable),en
                          el registro A obtenemos:
                          FF= No tecla     F7= Right Grph
                          FE= Shift        EF= Escape
                          FD= Ctrl         DF= Stop
                          FB= Left Grph    BF= Enter
```


...Ajedrez...

Tenemos el gusto de publicar un programa de ajedrez, gentileza de uno de nuestros colaboradores. Está escrito para PC's y Compatibles, aunque es posible adaptarlo a cualquier ordenador ya que no utiliza ninguna rutina externa al lenguaje BASIC

```
0    REM
100 CLS
110 A$=STRING$(80,205)
120 PRINT A$
130 PRINT TAB(34)"***  CHESS ***"
140 COLOR 23,0,0
150 PRINT :PRINT :PRINT TAB(39)"BANG"
160 COLOR 7,0,0
170 PRINT :PRINT :PRINT TAB(23)"FOR PERSONAL COMPUTERS  AND OTHERS"
180 PRINT :PRINT :PRINT TAB(19)"Pirate Soft. Company Ltd. (UNPROTEC SOFTWARE )"
190 PRINT A$
200 PRINT :PRINT :PRINT :PRINT :PRINT
210 PRINT TAB(28)"PRESS ANY KEY TO CONTINUE"
220 A$=INKEY$:IF A$="" THEN 220
230 CLS
240 CLEAR ,,2048
250 REM CHESS: MADRID, 1 DE DICIEMBRE DE 1986
260 REM ADAPTED FOR THE IBM PERSONAL COMPUTER.
270 DEFINT A-Z:A=0:B=0:X=0:Y=0:S=0:A0=0:T=0:A8=0:A1=0:A2=0:A3=0:A4=0:B1=0:B6!=0:
          H=0:M=0:N=0:P=0:W=0:A5=0:I=1:DIM C(64), A(10,10), B(10,10), T(10,10)
280 CLS:KEY OFF:INPUT "DO YOU WANT INSTRUCTIONS ";IN$:IN$=LEFT$(IN$,1):
    IF IN$="Y" THEN GOSUB 3500
290 CL$="N"
300 CLS:INPUT "YOUR NAME IS ";B$:IF B$="" THEN B$=" HUMAN " ELSE B$=LEFT$(B$,9)
310 CLS:PRINT "LEVEL OF PLAY";:INPUT B8!:IF B8!=0 THEN B8!=1:B7!=1+RND(0)/2:ELSE
    B7!=(B8!+1)/2+RND(0)/2
320 CLS:PRINT "DO YOU WANT WHITE ";B$;" ";:INPUT C$:GOSUB 1010:IF LEFT$(C$,1)<>"
N"   THEN GOSUB 1120:GOTO 430
330 GOSUB 1110:IF CL$="Y" THEN LOCATE 25,6,0:
     PRINT "START ";TIME$;
340 F!=-99:A0=0:FOR J=1 TO 8:FOR K=1 TO 8:IF A(J,K)=99 THEN A6=J:A7=K
350 NEXT:NEXT
360 FOR X=1 TO 8:FOR Y=1 TO 8:IF A(X,Y)<0 THEN GOSUB 500:
    IF F!>=B7! THEN GOTO 390
370 NEXT:NEXT:IF F!>=-9 THEN GOTO 390 ELSE GOSUB 1120
380 LOCATE 25,62,0: PRINT "I CONCEED    ";:GOTO 3470
390 A(R,U)=A(E,Q):A(E,Q)=1:IF A(R,U)=-2 AND U=1 THEN A(R,U)=-9
400 X=R:Y=U:A0=4:GOSUB 500:LOCATE 25,62,0:PRINT "IS MY MOVE    ";:
    LOCATE 25,51,0:PRINT CHR$(E+64);Q;"- "CHR$(R+64);U;
410 IF CL$="Y" THEN LOCATE 25,31,0:PRINT "STOP ";TIME$;
420 X=E:Y=Q:GOSUB 1130:X=R:Y=U:GOSUB 1130:IF C=1 THEN LOCATE 24,1,1:
      PRINT "CHECK ";:C=0 ELSE LOCATE 24,1,1:PRINT "       ";
430 IF X$="S" THEN GOSUB 3090:B7!=3*RND(0):LOCATE 25,1,1:PRINT "SELF-PLAYING";:
GOSUB 3800:GOTO 340:ELSE LOCATE 25,1,1:PRINT "MOVE "+CHR$(7);B$;:GOSUB 2850:D=0:
B4=A:B9=B:B5=0:IF A(X,Y)=2 AND Y=5 AND B=6 AND A(A,B)=1 AND ABS(A-X)=1 THEN B5=1
:GOTO 460
440 IF X$="K" OR X$="Q" THEN GOSUB 1120:LOCATE 25,1,0:PRINT STRING$(14," ");:
     GOTO 340
450 IF X$="S" THEN CL$="N":GOTO 430:ELSE A0=3:GOSUB 490:
    IF D=0 THEN LOCATE 25,31,0:PRINT "ILLEGAL       ";:GOTO 430
460 LOCATE 25,31,1:PRINT "OK                 ";:A=B4:B=B9:A(A,B)=A(X,Y):A(X,Y)=1:
    GOSUB 1130:X=A:Y=B:GOSUB 1130:IF N$="N" THEN GOSUB 1120:N$=" "
```

```
2870 REM DATOS DE LOS SPRITES 3P
2880 DATA 1,3,3,135,231,127,126,56,23,16,22,16,20,11,8,7,128,192,192,225,231,254
,126,25,232,8,104,8,40,208,16,224
2890 DATA 5,3,5,1,15,63,127,255,156,173,173,156,255,127,63,15,160,192,160,128,24
0,252,254,255,227,119,119,247,255,254,252,240
2900 DATA 199,152,166,198,206,79,32,31,21,42,74,82,84,148,164,0,227,25,101,99,11
,252,4,248,168,84,82,74,42,41,37,0
2910 REM ACABA LA PANTALLA
2920 SPRITEOFF:FOR T=1 TO 21:Y=Y+1:PUT SPRITE 1,(X,Y),9,1:NEXT
2930 FOR T=X TO 240STEP 5:PUT SPRITE 1,(T,Y),9,1:FOR A=1 TO 100:NEXT:PUT SPRITE
1,(T,Y),9,2:FOR A=1 TO 100:NEXT:NEXT:PLAY"L505S1M10000CDECR45CDECR45EFG3R45EFG3"
2940 RETURN
2950 REM CAIDA AL POZO
2960 Y=YL
2970 FOR T=1 TO 27:PUT SPRITE 1,(X,Y),9,1:Y=Y+1:NEXT:RETURN
2980 SCREEN 0
2990 PRINT"HAS ACABADO EL JUEGO"
3000 PRINT"CONSEGUISTE ";PU;" PUNTOS"
3010 PRINT"TE QUEDARON ";VI;" VIDAS"
3020 IF PU=<RC THEN PRINT"NO HAS BÀTIDO EL RECORD DE ";NN$;" CON ";RC;" PUNTOS"E
LSEGOTO 2740
3030 IF PA-1=1THENTER$=""ELSEIFPA-1>1THENTER$="S"ELSEIFPA-1=0THENTER$="S"
3040 PRINT"ATRAVESASTE ";PA-1;" CAVERNA"+TER$:GOTO 2840
3050 END
3060 REM INSTRUCCIONES
3070 CLS
3080 LOCATE 8,0:PRINT"/////////////////////////#
3090 LOCATE 8,1:PRINT"■ CAVERNAS LUNARES ■
3100 LOCATE 8,2:PRINT"#/////////////////////////
3110 FOR T=1 TO 2000:NEXT:CLS
3120 LOCATE 15,0:PRINT"1 PANTALLA"
3130 PRINT:PRINT:PRINT:PRINT:PRINT
3140 PRINT"ESTAS EN UNA CUEVA DONDE HAY UN CANDELA-BRO UNA ARANA Y UNA VALLA"
3150 PRINT"EL CANDELABRO SUBE Y BAJA DEBES PASARLO SIN QUE TE TOQUE AL IGUAL QUE
 LA VALLA HAY UN POZO QUE APARECE Y DESAPARECE  INTENTA NO CAERTE EN EL"
3160 PRINT"LA ARANA VA DE UN LADO A OTRO DEBES SE-GUIRLE HASTA LLEGAR AL AGUJERO
 DONDE   SALDRAS A OTRA PANTALLA"
3170 PRINT:PRINT:PRINT:PRINTTAB(10)"PULSA UNA TECLA"
3180 Z$=INPUT$(1)
3190 CLS
3200 PRINTTAB(15)"2 PANTALLA"
3210 PRINT:PRINT:PRINT:PRINT
3220 PRINT"TUS ENEMIGOS ESTAN EN UNA OSCURA CAVER- NA  DONDE POSEES LAMPARAS SEG
UN EL NI-VEL DE DIFICULTAD .DEBES ENCENDERLA PA-RA PODER VER TUS ENEMIGOS PERO S
OLO DU- RANTE UN DETERMINADO TIEMPO"
3230 PRINT"ENCIENDE LA LAMPARA CON LA BARRA DE ES-PACIO O EL DISPARADOR DEL JOYS
TICK"
3240 PRINT:PRINT:PRINTTAB(10)"PULSA UNA TECLA"
3250 Z$=INPUT$(1)
3260 CLS
3270 PRINTTAB(15)"3 PANTALLA"
3280 PRINT:PRINT:PRINT:PRINT
3290 PRINT"AHORA TUS ENEMIGOS SUBEN Y BAJAN DEL TE-CHO DE LA CAVERNA DEBES EVITA
RLOS AL   IGUAL QUE CAER EN LOS POZOS"
3300 PRINT"DEBES LLEGAR AL AGUJERO PARA LLEGAR A  LA SUPERFICIE LUNAR DONDE TE E
SPERA TU NAVE PARA REGRESAR A CASA"
3310 PRINT:PRINT:PRINT"          !!! S U E R T E !!!"
3320 PRINT:PRINT:PRINTTAB(5)"PULSA UNA TECLA PARA JUGAR"
3330 Z$=INPUT$(1)
3340 RETURN
3350 INPUT"DESEAS JUGAR DE NUEVO (S/N)";JUG$
3360 IF JUG$="S" THEN 210 ELSE IF JUG$="N"THEN ENDELSEIFJUG$<>"S"ORJUG$<>"N"THEN
 3350
3370 REM INICIALIZAR SPRITE 1P
3380 Y2=69:X1=240:Y3=15
3390 PUT SPRITE 4,(X3,Y3),15,5
3400 PUT SPRITE 2,(X1,Y1),15,3
3410 PUT SPRITE 3,(X2,Y2),15,4
3420 LINE(90,58)-(106,84),6,BF
3430 TR=0
3440 RETURN
3450 REM INICIALIZAR SPRITES 3P
3460 Y1=10
3470 PUT SPRITE 3,(X1,Y1),15,14
3480 PUT SPRITE 4,(X2,Y1),15,15
3490 PUT SPRITE 5,(X3,Y1),15,16
3500 LINE(40,58)-(56,84),6,BF:LINE(92,58)-(108,84),6,BF:LINE(146,58)-(162,84),6,
BF
3510 TR=0
3520 RETURN
3530 REM INICIALIZAR SPRITES 2P
3540 IF X=<80 THEN X=0:Y=42
3550 IF X=>80 THEN X=120:Y=62
3560 RETURN
```

JUAN CARLOS BUENO SANCHEZ
TORRES QUEVEDO, 21-4 B -SALAMANCA
EDAD 12 Años

■ Rutina para imprimir un numero (Byte) introducido en el registro A en su equivalente Hexadecimal:

```
HEX       F5     Push AF          ;Salvar Acumulador
        E6F0     And F0           ;Mascara 11110000 pasan altos
          1F     Rra              ;Correr a la derecha 4
          1F     Rra
          1F     Rra
          1F     Rra
        C630     Add A,30         ;Sumar 48, para ASCII
        FE3A     Cp A,3A
        3802     Jr C       NUM   ;Salta si es un numero
        C607     Add A,07         ;Suma 7 si letra
NUM       DF     Rst 18           ;Imprime en pantalla
          F1     Pop AF           ;Recupera Byte
        E60F     And 0F           ;Mscra 00001111 pasan bajos
        C630     Add A,30         ;Idem arriba
        FE3A     Cp A,3A
        3802     Jr C       NUM1
        C607     Add A,07
NUM1      DF     Rst 18
        3E20     Ld A,20          ;Imprime un espacio
          DF     Rst 18
          C9     Ret              ;Retorno
```

Ejemplo de utilización de la rutina arriba indicada:

```
KHEX CD3D40     Call 403D        ;Leer del teclado
     CDB8D2     Call HEX         ;Imprimir Byte Hexadecimal
       18F8     Jr KHEX          ;Salta al principio
```

Listado BASIC:

```
1 CLEAR100,&HD2B8:RESTORE5
2 FORI=&HD2B8TO&HD2DF:READA$:POKEI,VAL("&H"+A$):NEXTI
3 DEFUSR0=&HD2D8
4 A=USR0(0)
5 DATA f5,e6,f0,1f,1f,1f,1f,c6,30,fe,3a,38,2,c6,7,df,f1,e
6,f,c6,30,fe,3a,38,2,c6,7,df,3e,20,df,c9,cd,3d,40,cd,b8,d
2,18,f8
```

■ Rutina para utilizar ordenador e impresora como una maquina de escribir:

```
MAQ     3E00    Ld A,00         ;Para imprimir en pantalla
      3242F5    Ld(F542),A
      CD3D40    Call 403D       ;Leer teclado
          DF    Rst 18          ;Imprime caracter
      2142F5    Ld HL,F542      ;Para imprimir impresora
        CBC6    Set 0,(HL)
          DF    Rst 18          ;Imprime caracter
        18EF    Jr MAQ          ;Salta inicio
```

Listado BASIC:

```
1 CLEAR100,&HD2B8:RESTORE5
2 FORI=&HD2B8TO&HD2C8:READA$:POKEI,VAL("&H"+A$):NEXTI
3 DEFUSR0=&HD2B8
4 A=USR0(0)
5 DATA 3e,0,32,42,f5,cd,3d,40,df,21,42,f5,cb,c6,df,18,ef
```

■ Rutina para imprimir en pantalla o impresora todos los caracteres:

```
            C5    Push BC         ;Salvar BC
            F5    Push AF         ;Salvar Acumulador
          3E01    Ld A,00         ;00 Pantalla ,01 Impresora
        3242F5    Ld(F542),A
          060A    Ld B,0A         ;10 lineas
CHR         C5    Push BC
          065F    Ld B,5F         ;95 Caracteres
          3E20    Ld A,20         ;Print espacio
CHR1        DF    Rst 18          ;Imprime caracteres
            3C    Inc A
          10FC    Djnz CHR1
          3E0D    Ld A,0D         ;CR Retorno carro
            DF    Rst 18
          3E0A    Ld A,0A         ;LF Descender una linea
            DF    Rst 18
            C1    Pop BC          ;Recupera para 10 lineas
          10EE    Djnz CHR
            F1    Pop AF          ;Recupera Acumulador
            C1    Pop BC          ;Recupera BC
            C9    Ret             ;Retorno
```

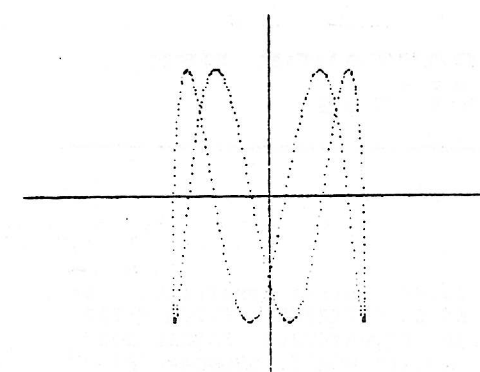
Listado BASIC:

```
1 CLEAR100,&HD2B8
2 FORI=&HD2B8TO&HD2D5:READA$:POKEI,VAL("&H"+A$):NEXTI
3 DEFUSR0=&HD2B8
4 A=USR0(0)
5 DATA c5,f5,3e,0,32,42,f5,6,A,c5,6,5f,3e,20,df,3c,10,fc
,3e,d,df,3e,a,df,c1,10,ee,f1,c1,c9
```



```
2120 REM CAIDA AL POZO
2130 Y=CP:FOR I=1TO27:PUT SPRITE 1,(X,Y),9,1:Y=Y+1:NEXT:RETURN
2140 REM SUMA BONOS
2150 LINE(80,140)-(150,150),4,BF:LOCATE 80,140:PRINTUSING"###,###";PU:SOUND 8,16
:SOUND 7,218:SOUND 9,16:SOUND10,16:SOUND11,16:SOUND 12,16:SOUND13,0
2160 FOR T=1 TO 2000:NEXT
2170 RETURN
2180 REM BORRA SPRITES
2190 PUT SPRITE 1,(X,Y),0,0:PUT SPRITE2,(170,100),0,0:PUT SPRITE 3,(40,45),0,0:P
UT SPRITE 4,(120,26),0,0:PUT SPRITE 5,(178,59.,0,0:PUT SPRITE 6,(221,42),0,0:PUT
 SPRITE 7,(200,22),0,0:PUT SPRITE 8,(80,62),0,0
2200 LINE(0,0)-(256,80),4,BF
2210 LINE(0,0)-(256,18),6,BF:LINE(0,59)-(256,85),6,BF:FOR T=0 TO256STEP13:CIRCLE
(T,15),10.4,,,2:PAINT(T,16),4:NEXT
2220 LINE(204,58)-(220,79),5,BF:LINE(220,63)-(256,79),5,BF
2230 X=0:Y=42:X1=78:Y1=10:X2=130:X3=188
2240 TP=0
2250 GOSUB 1770
2260 REM SPRITES 3P
2270 RESTORE 2880:FOR SP=14TO16:FOR BI=1TO32:READ A:S$=S$+CHR$(A):NEXT
2280 SPRITE$(SP)=S$:S$=""
2290 NEXT
2300 MH=0
2310 REM COMIENZO
2320 SPRITE ON
2330 D=STICK(ST)
2340 IF POINT(X+8,Y+16)=1 THEN SPRITE OFF:CP=40:GOSUB 2120:GOSUB 900:GOTO 2330
2350 IF POINT(X+8,Y+16)=3 THEN SPRITEOFF:CP=92:GOSUB 2120:GOSUB 900:GOTO 2330
2360 IF POINT(X+8,Y+16)=12 THEN SPRITEOFF:CP=146:GOSUB 2120:GOSUB900:GOTO 2330
2370 IF TT=>0 THEN TR=TR+1
2380 IF TR=TT THEN LINE(40,58)-(56,84),1,BF:LINE(92,58)-(108,84),3,BF:LINE(146,5
8)-(162,84),12,BF
2390 IF TR=TS THEN LINE(40,58)-(56,84),6,BF:LINE(92,58)-(108,84),6,BF:LINE(146,5
8)-(162,84),6,BF:TR=0
2400 IF POINT(X,Y+16)=5 THEN SPRITEOFF:GOSUB 2910:GOSUB 1950:GOSUB 2140:GOTO 256
0
2410 ON SPRITE GOSUB 900
2420 IF D=3 THEN MH=1
2430 IF D=7 THEN MH=2
2440 IF MH=1 THEN X=X+2
2450 IF MH=2 THEN X=X-2
2460 IF X=<0THEN X=0
2470 IF Y1=<10THENYH=1
2480 IF Y1=>42THENYH=2
2490 IF YH=1 THEN Y1=Y1+VJ
2500 IF YH=2 THEN Y1=Y1-VJ
2510 PUT SPRITE 1,(X,Y),9,1
2520 PUT SPRITE 3,(X1,Y1),2,14
2530 PUT SPRITE 4,(X2,Y1),10,15
2540 PUT SPRITE 5,(X3,Y1),3,16
2550 GOTO 2330
2560 REM FINAL DEL JUEGO
2570 RESTORE2570:CLS:FOR SP=17TO18:FOR BI=1 TO 32:READ A:S$=S$+CHR$(A):NEXT:SPRI
TES(SP)=S$:S$="":NEXT:DATA 3,6,12,24,50,102,196,192,255,255,255,255,39,79,79,224
,192,96,48,88,204,150,51,3,255,255,255,255,228,242,242,7,0
2580 DATA 15,15,15,15,15,7,7,7,7,7,3,3,3,1,1,1,240,240,240,240,240,224,224,224,2
24,224,192,192,192,128,128,128
2590 CIRCLE(128,300),200,13:PAINT(128,190),13:PUT SPRITE 1,(230,150),7,17
2600 FOR C=1 TO 200:A=INT(RND(1)*256)+1:B=INT(RND(1)*90):Q=INT(RND(1)*16):PSET(A
,B),Q:NEXT
2610 PLAY"O5L12S1M10000CDEC9CDEC9EFG5EFG5L20GAGFE12C12L20GAGFE12C12D12O4B10O5C3L
12O6CDEC9CDEC9EFG5EFG5L20GAGFE12C12L20GAGFE12C12D12O5B10O6C3"
2620 FOR T=0 TO 220 STEP 10
2630 FOR A=1 TO 150:NEXT
2640 PUT SPRITE 3,(T,150),9,1:FOR A=1 TO 150:NEXT:PUT SPRITE 3,(T,150),9,2:NEXT
2650 PUT SPRITE 3,(0,0),0,1:PUT SPRITE 3,(0,0),0,2
2660 FOR T=150 TO 0 STEP -1
2670 PUT SPRITE 1,(230,T),7,17:PUT SPRITE 2,(230,T+16),8,18:FORR=1 TO50:NEXT:SOU
ND 8,8:SOUND 0,T:NEXT:SOUND8,0
2680 CLS
2690 SCREEN 0
2700 PRINT"Muy bien ";NO$
2710 PRINT"Has llegado a la superficie Lunar.Has  pasado tres cavernas en el niv
el de dificultad ";NV
2720 PRINT"Has conseguido una puntuacion de ";PU;"puntos
2730 IF PU=>RC THEN GOTO 2750
2740 PRINT:PRINT"Lo siento no has conseguido superar el record de ";NN$;" con ";
RC;" puntos":GOTO 2840
2750 PRINT"Pulsa ENTER"
2760 K$=INKEY$:IF K$=CHR$(13)THEN2770ELSE2760
2770 REM HAS CONSEGUIDO RECORD
2780 CLS:PRINTTAB(5)" !!! F E L I C I D A D E S !!!"
2790 PRINT"Has conseguido superar el record de    ";NN$;" con";RC;"puntos"
2800 PRINT:PRINT:PRINTTAB(10)"EL NUEVO RECORD ES..."
2810 RC=PU:NN$=NO$
2820 FOR T=1 TO 2000:NEXT
2830 PRINTTAB(10);NN$;":";RC
2840 FOR T=1 TO7000:NEXT:CLS
2850 INPUT"QUIERES JUGAR OTRA VEZ (S/N)";JUG$
2860 IF JUG$="S"THEN 210 ELSE IF JUG$="N"THENENDELSEIF JUG$<>"S"OR JUG$<>"N"THEN
 2850
```

```
1380 RESTORE 1430:FOR SP=1OTO 13
1390 FOR BI=1 TO 32
1400 READ A:S$=S$+CHR$(A)
1410 NEXT:SPRITE$(SP)=S$:S$="":NEXT
1420 REM DATAS DE LOS SPRITES
1430 DATA 3,7,15,31,19,33,33,51,127,113,112,120,127,121,48,0,192,224,240,248,200
,132,132,204,254,142,14,30,254,254,194,0
1440 DATA 5,3,5,1,1,7,31,255,197,236,237,237,127,63,31,7,64,128,64,0,0,192,240,2
54,162,182,54,182,252,248,240,192
1450 DATA 0,0,0,3,7,236,120,232,8,63,63,127,128,170,128,127,56,64,128,128,192,98
,100,100,104,232,232,252,2,170,2,254
1460 DATA 1,1,1,1,1,1,7,63,127,252,224,192,192,96,60,7,128,128,128,128,128,128,2
24,252,254,63,7,3,3,6,60,224
1470 REM PRESENTA SPRITES OCULTOS
1480 PUT SPRITE 2,(100,170),10,13:LOCATE 125,175:PRINT"#";LM:PUT SPRITE3,(40,45)
,1,10:PUT SPRITE 4,(120,26),1,11:PUT SPRITE 5,(178,59),1,12:PUT SPRITE 6,(221,42
),1,3:PUT SPRITE 7,(200,22),1,5:PUT SPRITE 8,(80,62),1,4
1490 SPRITEON
1500 REM COMIENZO
1510 X=0:Y=42
1520 D=STICK(ST)
1530 IF STRIG(ST)=-1 THEN GOSUB2040
1540 ON SPRITE GOSUB 900
1550 IF D=1 THEN XH=1
1560 IF D=2 THEN XH=2
1570 IF D=3 THEN XH=3
1580 IF D=4 THEN XH=4
1590 IF D=5 THEN XH=5
1600 IF D=6 THEN XH=6
1610 IF D=7 THEN XH=7
1620 IF D=8 THEN XH=8
1630 IF  XH=1 THEN Y=Y-4
1640 IF XH=2 THEN X=X+4:Y=Y-4
1650 IF XH=3 THEN X=X+4
1660 IF XH=4 THEN X=X+4:Y=Y+4
1670 IF XH=5 THEN Y=Y+4
1680 IF XH=6 THEN Y=Y+4:X=X-4
1690 IF XH=7 THEN X=X-4
1700 IF XH=8 THEN X=X-4:Y=Y-4
1710 IF X=<0 THEN X=0
1720 IF Y=<20THENY=20
1730 IF Y=>64 THEN Y=64
1740 IF X=>224THENPUT SPRITE 2,(100,170),0,0:LINE(75,175)-(150,185),4,BF:LINE(1,
21)-(239,79),15,BF:PLAY"L5O5S1M10000CDECR45CDECR45EFG3R45EFG3":GOSUB1950:GOSUB21
40:GOTO 2180
1750 PUT SPRITE 1,(X,Y),9,1:FOR R=1TO75:NEXT:PUT SPRITE 1,(X,Y),9,2:FORR=1TO75:N
EXT
1760 GOTO 1520
1770 REM PANEL DE PUNTOS Y VIDAS
1780 PA=PA+1
1790 FOR T=90 TO 180STEP35
1800 LINE(20,T)-(246,T+30),4,BF:NEXT
1810 LOCATE 25,95:PRINT"ENERGIA:"
1820 LINE(79,93)-(232,103),1,B
1830 FOR T=80TOEN:LINE(T,94)-(T,102),7:NEXT
1840 ES=EN-80
1850 FOR C=80 TO 230 STEP 2
1860 LINE(C,105)-STEP(0,2),1:NEXT
1870 FOR C=80 TO 230 STEP 10
1880 LINE(C,105)-STEP(0,5),1:NEXT
1890 LOCATE 25,112:PRINT"ENERGIA:":LOCATE 80,112:PRINTES
1900 LOCATE 25,130:PRINT"VIDAS:";V
1910 LOCATE 25,140:PRINT"PUNTOS:":LOCATE 80,140:PRINTUSING"###,###";PU
1920 LOCATE 25,165:PRINT"PANTALLA:";PA
1930 LOCATE 25,175:PRINT"NIVEL";NV
1940 RETURN
1950 REM SUMA PUNTOS
1960 FOR C=1 TO 4700:NEXT
1970 FOR T=ENTO80STEP-1:SOUND8,8:LINE(T,94)-(T,102),4:PU=PU+10:PU=PU+T*2:SOUND0,
T:LINE(80,140)-(150,150),4,BF:LOCATE 80,140:PRINTUSING"###,###";PU:LINE(80,112)-
(105,120),4,BF:LOCATE 80,112:PRINTES:ES=ES-1
1980 NEXT:SOUND8,0
1990 LOCATE 90,130:PRINT"* 50 =";V*50:PU=PU+V*50
2000 LOCATE 80,175:PRINT"* 50 =";NV*50:PU=PU+NV*50
2010 LOCATE 100,165:PRINT"* 50 =";PA*50:PU=PU+PA*50
2020 FOR T=1 TO 1000:NEXT
2030 RETURN
2040 REM LAMPARA
2050 LM=LM-1
2060 IF LM<0 THEN RETURN
2070 LINE(125,175)-(145,185),4,BF:LOCATE 125,175:PRINT"#";LM
2080 LINE(1,21)-(239,79),15,BF
2090 FOR T=1 TO 150:SOUND8,8:SOUND 0,T+100/3:NEXT:SOUND8,0:LINE(1,21)-(239,79),1
,BF
2100 GOSUB 1000
2110 RETURN
```

SPECTRAVIDEO

■ Rutina alta resolución en metodo grafico para copiar pantalla en modo 1 (Screen 1) a impresora. Si los graficos contienen colores ATENCION, puede ser que no salga todo o lo que salga quede ligeramente modificado. Es un poco complicado explicar aqui la solución pero creo que valdra la pena esperar a que ultime otra rutina con gamas de grises (más o menos puntitos) y más grande.

Listado BASIC:

```
22000 CLEAR500,&HD2B8:RESTORE22025
22005 FORI=&HD2B8TO&HD368:READA$:POKEI,VAL("&H"+A$):NEXTI
22010 DEFUSR0=&HD2D9
22015 PRINTUSR0(0)
22025 DATA 1B,41,8,1B,6C,A,0,1B,4B,0,1,7E,DF,23,10,FB,C9,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,3E,1,32,42,F5,6,6,21,B8,D2,CD,C3,D2,21,
0,0,6,18,C5,E5,6,4,21,BF,D2,CD,C3,D2,E1,6,20,C5,6,8,11,D1,D2,CD,
34,37,A7,20,21,D5,11,0,20
22030 DATA 19,CD,34,37,A7,ED,52,D1,FE,7,28,A,FE,A,28,8,FE,E,28,4
,FE,F,20,4,3E,FF,18,1,AF,12,23,13,10,D4,11,D8,D2,E5,21,C9,D2,E5
22035 DATA 6,8,C5,D5,6,8,E5,D5,E1,CB,6,E1,CB,1E,1B,10,F5,23,D1,C
1,10,EC,E1,6,8,7E,FE,9,20,1,3D,DF,23,10,F6,E1,C1,10,9F,3E,D,DF,3
E,A,DF,C1,10,89,3E,1B,DF,3E,40,DF,C9
```

Grabar en disco con SAVE"1:M1COPY",A. Para utilizarla cargar primero el programa que tiene la pantalla que queremos copiar, y despues, cargamos con MERGE"1:M1COPY la rutina, y realizamos una llamada RUN 22000 etc. Si queremos podemos trozear la rutina e introducirla donde mejor nos convenga, realizando una llamada PRINT USR0(0), ejecutamos la copia a impresora.

MSX MSX MSX MSX MSX MSX MSX

ADAPTADOR MSX para SVI 328

Sabia usted que su SVI328 puede correr programas tan populares como: **ZAXXON, CHESS86, ALIEN 8, GUNFRYGHT, FLYPATH-737, GHOST BUSTERS, CENTIPEDE, KINGHT LORE, PROFANATION, PASCAL SONY, SORCERY, BUCK ROGERS, HERO, MAZIACS, MUTANT MONTY, SNOWMAN, NIGHT SHADE, OH SHIT, ULTRA CHESS, ZEN, BUGABOO** etc, etc, etc... ?

Desde ahora esto es posible gracias al **ADAPTADOR MSX** para SVI328 . Este adaptador le permitirá disfrutar de todos los programas que hemos mencionado arriba y de varios centenares más que podrá usted ir descubriendo en el catálogo de programas MSX.

El **ADAPTADOR MSX** para SVI328 es, en esencia, un sistema software capaz de gestionar su SVI328 como si se tratara de un MSX real. Sus excepcionales caracteristicas lo destacan decididamente del adaptador **MSX** fabricado por Spectravideo, puesto que resuelve todos los problemas que planteaba su utilizacion y presenta caracteristicas nuevas.

Podemos resumir brevemente sus caracteristicas en los siguientes puntos:

- **BASIC compatible MSX**. Todos los programas BASIC MSX corren sobre el SVI328+ADAPTADOR. Todos los trucos que pueden utilizarse en un MSX funcionan perfectamente con nuestro adaptador.

- Memoria disponible en BASIC: 28815 bytes.
  Memoria para programas en código de máquina: 48k.
  Se mantienen reservas sobre el funcionamiento de programas que efectúen chequeo de carga o carguen los programas de forma extraña al MSX(en especial "Turbo Load" y carga "bit a bit").

- **Teclado completamente compatible MSX** con acentos y todos sus caracteres gráficos. El teclado numèrico mantiene todas sus funciones.

- **Completa compatibilidad MSX en grabaciones sobre cassette.** Los programas BASIC o C.M. pueden grabarse y cargarse a cualquiera de las 2 velocidades estàndar MSX: 1200 o 2400 baudios. La intercambiabilidad entre los programas grabados con nuestro ADAPTADOR y los programas MSX es total.


```
680 IF D=7 THEN MH=2
690 IF MH=1 THEN X=X+2
700 IF MH=2 THEN X=X-2
710 IF X=<0THENX=0
720 IF X=>240THENEND
730 IF Y2=<42THENYH=1
740 IF Y2=>69THENYH=2
750 IF YH=1 THEN Y2=Y2+VJ
760 IF YH=2 THEN Y2=Y2-VJ
770 IF X1=>240THENXH=1
780 IF X1=<170 THEN XH=2
790 IF XH=1 THENX1=X1-VJ
800 IF XH=2 THENX1=X1+VJ
810 IF Y3=>42THENYY=1
820 IF Y3=<15THENYY=2
830 IF YY=1 THEN Y3=Y3-VJ
840 IF YY=2 THEN Y3=Y3+VJ
850 PUT SPRITE 4,(X3,Y3),14,5
860 PUT SPRITE 1,(X,Y),9,1
870 PUT SPRITE 2,(X1,Y1),7,3
880 PUT SPRITE 3,(X2,Y2),10,4
890 GOTO 600
900 REM EXPLOSION DE HOMBRE
910 IF PA=1 THEN GOSUB 3370
920 IF PA=3 THEN GOSUB 3450
930 PUT SPPITE 1,(X,Y),10,6:SOUND8,16:SOUND7,249:SOUND9,16:SOUND10,16:SOUND11,16
:SOUND12,16:SOUND13,0:SOUND8,0
940 FOR C=1 TO500:NEXT:PUT SPRITE 1,(X,Y),10,7:FOR C=1 TO200:NEXT:SOUND7,234:SOU
ND8,16:SOUND9,16:SOUND10,16:SOUND11,16:SOUND12,54:SOUND13,0:SOUND8,0:PUT SPRITE
1,(X,Y),10,8:FOR C=1 TO100:NEXT:PUT SPRITE 1,(X,Y),10,9
950 FOR T=9 TO 6STEP-1
960 PUT SPRITE 1,(X,Y),10,T:FOR C=1 TO200:NEXT:NEXT:IF PA=1 THEN Y=42 ELSE IF PA
=3 THEN Y=42
970 IF PA=2 THEN GOSUB 3530
980 SPRITEOFF
990 X=X-20:PUT SPRITE 1,(X,Y),10,1
1000 ES=EN-80
1010 FOR T=1 TO 10
1020 LINE(EN,94)-(EN,102),4:EN=EN-1:ES=ES-1:LINE(80,112)-(120,120),4,BF:LOCATE 8
0,112:PRINTES:IF EN=<80THEN1060ELSENEXT
1030 IF V=<0THEN2980
1040 MH=0
1050 SPRITEON:RETURN
1060 REM QUITA UNA VIDA
1070 V=V-1
1080 LINE(60,129)-(80,137),4,BF:LOCATE 60,130:PRINTV
1090 IF V=<0THEN2980
1100 EN=230
1110 FOR T=80TO230
1120 LINE(T,94)-(T,102),7
1130 NEXT:EN=230:ES=EN-80:LINE(80,112)-(140,120),4,BF:LOCATE 80,112:PRINTES
1140 SPRITEON:RETURN
1150 REM DATAS DE LOS SPRITES 1P
1160 DATA 7,4,4,7,7,15,31,27,31,15,7,7,14,124,120,96,192,128,128,128,192,64,56,1
32,248,192,192,224,112,112,48,56
1170 DATA 7,4,4,7,7,15,31,27,31,15,1,1,1,1,1,1,192,128,128,128,192,64,56,132,248
,224,224,128,128,128,128,192
1180 DATA 1,1,1,25,125,255,121,57,63,30,30,15,12,25,63,51,128,128,128,152,190,25
5,158,156,252,120,120,240,48,152,252,204
1190 DATA 1,3,49,121,49,49,49,49,57,63,31,1,1,13,63,127,128,192,140,158,140,140,
140,140,156,252,248,128,128,176,252,254
1200 DATA 36,127,36,127,36,127,36,127,36,127,36,127,36,127,36,0,146,255,146,255,
146,255,146,255,146,255,146,255,146,255,146,0
1210 DATA 0,0,0,0,0,1,3,15,31,15,3,1,0,0,0,,0,0,0,0,0,128,192,240,248,240,192,12
8,0,0,0,0
1220 DATA 0,0,0,1,3,3,15,31,63,31,15,3,3,1,0,0,0,0,0,128,192,192,240,248,252,248
,240,192,192,128,0,0
1230 DATA 0,0,1,3,3,7,15,63,127,63,15,7,3,3,1,0,0,0,128,192,192,224,240,252,254,
252.240,224,192,192,128,0
1240 DATA 0,1,1,3,3,7,15,127,255,127,15,7,3,3,1,1,0,128,128,192,192,224,240,254,
255,254,240,224,192,192,128,128
1250 REM ACABA PANTALLA
1260 MH=0:XH=0
1270 GOSUB 2910
1280 GOSUB 1950:GOSUB 2140
1290 REM CAMBIA LA PANTALLA
1300 LINE(0,0)-(256,85),4,BF
1310 SPRITEOFF:PUT SPRITE 1,(X,Y),0,0:PUT SPRITE3,(X1,Y1),0,0:PUT SPRITE2,(X2,Y2
),0,0:PUT SPRITE 4,(X3,Y3),0,0
1320 REM DIBUJO DE LA 2P
1330 N=RND(-TIME):LINE(0,0)-(256,85),1,BF:Y=20:FOR T=0TO256:A=INT(RND(1)*150):IF
 A=<50 THENY=Y+1ELSEIFA=>50ANDA=<100THENY=Y-1ELSEIF A=>100 THEN Y=Y+0
1340 LINE(T,Y)-(T,0),15:IF Y=>19THENY=Y-1
1350 NEXT
1360 GOSUB 1770
1370 REM SPRITES 2P
```

CAVERNAS LUNARES

Felicitamos a Juan Carlos Bueno de Salamanca porque sólo tienen 12 años y es el autor de este programa que publicamos a continuación.

```
10 REM ▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓
20 REM ■ Juan Carlos Bueno Sanchez ■
30 REM ■       Cavernas Lunares      ■
40 REM ■      Para Club I.S.I.S.     ■
50 REM ■            -1986-           ■
60 REM ▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓
70 REM
80 REM ▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓
90 REM ■ Presentacion ■
100 REM ▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓
110 REM
120 COLOR 2,1,9
130 SCREEN 2,2
140 FOR T=1 TO 8:READ A,C$:LOCATE A,50:PRINTC$:SOUND8,16:SOUND7,234:SOUND9,16:SO
UND10,16:SOUND11,16:SOUND12,16:SOUND13,0:FOR Q=1 TO 75:SOUND 8,8:SOUND 0,100-Q:N
EXT:SOUND8,0
150 NEXT
160 FOR T=1 TO 7:READ A,C$:LOCATE A,80:PRINTC$:SOUND8,16:SOUND7,234:SOUND9,16:SO
UND10,16:SOUND11,16:SOUND12,16:SOUND13,0:FOR Q=1 TO 75:SOUND8,8:SOUND0,100-Q:NEX
T:SOUND 8,0
170 NEXT
180 FOR T=0 TO 255:PSET(T,32),3:PSET(T,125),3:SOUND8,8:SOUND0,T:NEXT:SOUND8,0
190 FOR R=1 TO 500:NEXT
200 RC=115000!:NN$="Svi Personal Computer"
210 INPUT"POR FAVOR INTRODUCE TU APODO O NOMBRE";NO$
220 PA=0
230 DATA 10,C,40,A,70,V,100,E,130,R,160,N,190,A,220,S
240 DATA 20,L,50,U,80,N,110,A,140,R,170,E,200,S
250 CLS
260 INPUT"NIVEL DE DIFICULTAD 1-PRINCIPANTE                          2-AFICIONAD
O                            3-EXPERTO";NV
270 IF NV<1ORNV>3THEN250
280 IF NV= 1  THEN VJ=1:LM=4:TT=20:TS=30
290 IF NV=2 THEN VJ=1.5:LM=3:TT=20:TS=40
300 IF NV=3 THEN VJ=2:LM=2:TT=16:TS=30
310 PRINT"UTILIZA MAYUSCULAS POR FAVOR
320 INPUT"UTILIZAS JOYSTICK (S/N)";JOY$
330 IF JOY$="" THEN 320
340 IF JOY$="S" THEN ST=1:GOTO 360ELSEIFJOY$="N"THENST=0:GOTO 360ELSE320
350 IF JUG$<>"S"OR JUG$<>"N"THEN320
360 INPUT"DESEAS LEER LAS INSTRUCCIONES (S/N)";IN$
370 IF IN$="S"THENGOSUB3060ELSEIFIN$="N"THEN390ELSEIF IN$<>"S"ORN$<>"N"THEN360
380 IF IN$="" THEN 360
390 COLOR 9,1,14:SCREEN 1,2
400 REM SPRITES
410 RESTORE 1160
420 FOR SP=1 TO 9
430 FOR RI=1 TO 32
440 READ A:S$=S$+CHR$(A)
450 NEXT:SPRITE$(SP)=S$:S$="":NEXT
460 LINE(0,0)-(256,85),4,BF
470 PU=0:V=1:EN=230
480 GOSUB 1770
490 REM DIBUJO DE LA 1P
500 LINE(0,0)-(256,18),6,BF
510 LINE(0,59)-(256,85),6,BF
520 LINE(50,61)-(65,85),4,BF
530 FOR T=0TO256STEP13:CIRCLE(T,15),10,4,,,2:PAINT(T,16),4:NEXT
540 LINE(204,58)-(220,79),5,BF:LINE(220,63)-(256,79),5,BF
550 X1=170:Y1=42:X2=50:Y2=69
560 X=0:Y=42:X3=140:Y3=15:CP=0
570 TR=0
580 REM COMIENZO
590 SPRITEON
600 D=STICK(ST)
610 IF POINT(X+8,Y+16)=1THENSPRITEOFF:CP=90:GOSUB2120:GOSUB900:GOTO 600
620 IF TT=>0THENTR=TR+1
630 IFTP=TT THEN LINE(90,56)-(106,84),1,BF
640 IF TR=TS THEN LINE(90,58)-(106,84),6,BF:TR=0
650 IF POINT(X,Y+16)=5THENSPRITEOFF:GOSUB1250
660 ON SPRITE GOSUB 900
670 IF D=3 THEN MH=1
```


- No es necesario en ningún caso extraer el ADAPTADOR de su slot para pasar a BASIC SVI328 o CP/M puesto que se incorporan los comandos necesarios para gobernar todo el sistema desde el propio ADAPTADOR MSX.(No obstante puede quitarse cuando se desee puesto que se conecta en el slot de cartuchos). Es interesante el hecho de que pueden pasarse programas desde BASIC MSX a BASIC SVI328 y viceversa.

- El cassette, impresora, los joysticks y la unidad de discos pueden utilizarse sin problemas desde el ADAPTADOR.

- Se dispone opcionalmente de sistema de disco propio del ADAPTADOR, compatible CP/M. Esta compatibilidad permite el paso de programas entre MSX DOS y CP/M a través de un SVI728 al mismo tiempo que permite utilizar al completo las unidades de doble cara y mejora los tiempos de grabación y carga en mas del 80%. Además permite el uso de cualquier programa de utilidad CP/M para revisar programas MSX, crear código de máquina con macroensambladores,etc.

- El ADAPTADOR, aunque se comprueba antes de su salida, dispone de una garantía de 6 meses a partir de la fecha de compra. Se incluye manual de instrucciones con algunos comentarios útiles referentes al propio ADAPTADOR y a la norma MSX.

El precio al 1 de Noviembre del 86 es de:

7840 pts(IVA incluido).

Pídanos tarifas especiales para cantidades superiores a 10 unidades o si es usted distribuidor o detallista.

Remitir toda correspondencia a:

J.GONZALO
c/CRISTOBAL MELLO 11.4
PORTUGALETE
-VIZCAYA-

TFNO:(94) 462 46 52 (de 19h-21.30h)

IMPRIMIENDO....

Como podeis comprobar, esta carta la estoy escribiendo con el programa que os envio. Este programa està escrito para una impresora "COMPUTE MATE-100, CPA 80".

Espero que este programa solucione los problemas que tenia el socio del boletin nùmero uno.

INSTRUCCIONES para utilizar el programa:

En principio podemos seleccionar el juego de caracteres "internacionales", que deseemos.

Podemos utilizar en el mismo renglòn letra tipo pica, junto con letra tipo enfatizada.

Tambièn podemos utilizar comprimido junto con comprimido doble repicado.

No es aconsejable mezclar los tipos ELITE y EXPANDIDO entre ellos ni con otros, debido a la diferencia de tamaño de èstos.

Se puede subrayar cualquier palabra o frase, pulsando ^S antes de escribirla, pero al finalizar hay que pulsar de nuevo ^S para anular dicha funciòn.

Podemos anular el detector de papel de la impresora pulsando ^B.

El acento (') y la dièresis (¨), hay que pulsarlos antes que la letra sobre la que van.

Las ventanas de las teclas de funciòn nos indican las teclas que hay que pulsar para seleccionar el tipo de escritura, y las funciones de que disponemos.

La ventana idquierda (correspondiente a F1), nos recuerda en todo momento el tipo de escritura que estamos utilizando, asì como el subrayado, si bien al anular dicha funciòn la ventana se quedarà sin mensaje, aunque la impresora sigue escribiendo en el tipo de escritura seleccionado con anterioridad.

Funciones y letras especiales:

- ^P Escritura tipo Pica
- ^E " " Elite
- ^C " " Comprimido
- ^F " " Enfatizado
- ^D " " Comprimido doble repicado
- ^X " " Expandido
- ^S Subrayado
- ^B Anula detector de papel
- ESC Sale del progama (Fin)
- ENTER para el siguiente renglòn
- ñ LEF GRPH + "["
- Ñ "\"
- ¡ "["
- ¿ "]"
- ¨ "{"
- EL sìmbolo ^ corresponde a la tecla CTRL


```
0078 ENT      SETTXT
     ;
     ; Name:          SETT32
     ; Function:      Sets the VDP for text (32*24) mode.
     ; Entry:         T32NAM, T32CGP, T32COL, T32ATR, T32PAT
     ; Returns:       None
     ; Modifies:-     All
     ;
007B ENT      SETT32
     ;
     ; Name:          SETGRP
     ; Function:      Sets the VDP for high-resolution mode.
     ; Entry:         GRPNAM, GRPCGP, GRPCOL, GRPATR, GRPPAT
     ; Returns:       None
     ; Modifies:      All
     ;
007E ENT      SETGRP
     ;
     ; Name:          SETMLT
     ; Function:      Sets the VDP for multicolor mode.
     ; Entry:         MLTNAM, MLTCGP, MLTCOL, MLTATR, MLTPAT
     ; Returns:       None
     ; Modifies:      All
     ;
0081 ENT      SETMLT
     ;
     ; Name:          CALPAT
     ; Function:      Returns address of sprite pattern table.
     ; Entry:         Sprite ID in [Acc]
     ; Returns:       Address in [HL]
     ; Modifies:      AF, DE, HL
     ;
0084 ENT      CALPAT
     ;
     ; Name:          CALATR
     ; Function:      Returns address of sprite attribute table.
     ; Entry:         Sprite ID in [Acc]
     ; Returns:       Address in [HL]
     ; Modifies:      AF, DE, HL
     ;
0087 ENT      CALATR
     ;
     ; Name:          GSPSIZ
     ; Function:      Returns the current sprite size.
     ; Entry:         None
     ; Returns:       Sprite size (# of bytes) in [Acc]
     ;                Carry  set if  16*16 sprite in use,  otherwise
     ;                reset the otherwise.
     ; Modifies:      AF
     ;
008A ENT      GSPSIZ
     ;
     ; Name:          GRPPRT
     ; Function:      Prints a character on the graphic screen.
     ; Entry:         Code to output in [Acc]
     ; Returns:       None
     ; Modifies:      None
     ;
008D ENT      GRPPRT
```

```
0062 ENT      CHGCLR
     BOLE     1
     ;
     ; Name:             NMI
     ; Function:         Performs non-maskable interrupt procedures.
     ; Entry:            None
     ;
     ; Returns:          None
     ; Modifies:         None
     ;
0066 ENT      NMI
     ;
     ; Name:             CLRSPR
     ; Function:         Initializes all sprites.
     ;                   Patterns are set to nulls, sprite names are
     ;                   set to sprite plane number, sprite colors are
     ;                   set to foreground color, vertical positions
     ;                   are set to 209.
     ; Entry:            SCRMOD
     ; Returns:          None
     ; Modifies:         All
     ;
0069 ENT      CLRSPR
     ;
     ; Name:             INITXT
     ; Function:         Initializes screen for text mode (40*24) and
     ;                   sets the VDP.
     ; Entry:            TXTNAM, TXTCGP
     ; Returns:          None
     ; Modifies:         All
     ;
006C ENT      INITXT
     ;
     ; Name:             INIT32
     ; Function:         Initializes screen for text mode (32*24) and
     ;                   sets the VDP.
     ; Entry:            T32NAM, T32CGP, T32COL, T32ATR, T32PAT
     ; Returns:          None
     ; Modifies:         All
     ;
006F ENT      INIT32
     ;
     ; Name:             INIGRP
     ; Function:         Initializes screen for high-resolution mode
     ;                   and sets the VDP.
     ; Entry:            GRPNAM, GRPCGP, GRPCOL, GRPATR, GRPPAT
     ; Returns:          None
     ; Modifies:         All
     ;
0072 ENT      INIGRP
     ;
     ; Name:             INIMLT
     ; Function:         Initializes screen for multicolor mode and
     ;                   sets the VDP.
     ; Entry:            MLTNAM, MLTCGP, MLTCOL, MLTATR, MLTPAT
     ; Returns:          None
     ; Modifies:         All
     ;
0075 ENT      INIMLT
     ;
     ; Name:             SETTXT
     ; Function:         Sets the VDP for text (40*24) mode.
     ;
     ; Entry:            TXTNAM, TXTCGP
     ; Returns:          None
     ; Modifies:         All
     ;
```


También podemos utilizar los gráficos de que dispone la impresora, aprovechando los gráficos del ordenador, y pulsando las teclas correspondientes según la tabla adjunta.

La (L) y la (R) primera, corresponde a las teclas LEFT GRPH y RIGHT GRPH del ordenador, que hay que pulsar junto con la letra que tiene a continuación, para conseguir el gráfico que figura a su lado.

LQ┴ RQ♦ LW| RW♣ LE▬ RE♣ LT├ RTo LY┌ RY▒ LU‾ RU/ LI| RI‡ LO█ RO♠ LP┼ RP♥
LA_ RA└ LS┤ RS● LD▬ RD┐ LF■ RF┘ LG█ RG= LH█ RH├ LJ▌ RJ┤ LK▌ RK◢ LL▌ RL◣
LZ┐ RZ▓ LX| RX▨ LC▬ RC┌ LV— RV\ LB_ RB┘ LN█ RN◤ LM▌ RM◥ L ñ

TIPO PICA   tipo pica   **TIPO ENFATIZADO**   **tipo enfatizado**

TIPO COMPRIMIDO   tipo comprimido   **TIPO COMPRIMIDO DOBLE REPICADO**

T I P O   E X P A N D I D O   t i p o   e x p a n d i d o

TIPO ELITE   tipo elite

Carlos Garcia Dos Santos

```
10 '
20 '---- MAQUINA DE ESCRIBIR ----
30 '---- Carlos Garcia D.S. -----
40 '--------- julio (86) --------
50 '
60 CLEAR10000
70 CLS:COLOR15,5:SCREEN0,0
80 STOP ON:ON STOP GOSUB1090
90 LOCATE1,11:PRINT"█________________________________┐"
100 LOCATE1,12:PRINT"█ CONECTE LA IMPRESORA  Listo? (s/n) █"
110 LOCATE1,13:PRINT"┌┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┼┐"
120 LOCATE18,15:W$=INPUT$(1)
130 IF W$<>"s"ANDW$<>"S"THENBEEP:GOTO120
140 LPRINT CHR$(27);"9";
150 '----caracteres internacionales
160 CLS:RO$=CHR$(27)+"p":OF$=CHR$(27)+"q"
170 LOCATE0,0:PRINT"‾▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓ "
180 LOCATE0,1:PRINT"◣CARACTERES INTERNACIONALES ESPECIALES♠"
190 LOCATE0,2:PRINT"▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌▌ "
200 LOCATE12,4:PRINT "1   "+RO$+ "CASTELLANO"+OF$
210 LOCATE12,6:PRINT "2   "+RO$+ "U.S.A."+OF$
220 LOCATE12,8:PRINT "3   "+RO$+ "FRANCES"+OF$
230 LOCATE12,10:PRINT "4   "+RO$+ "ALEMAN"+OF$
240 LOCATE12,12:PRINT "5   "+RO$+ "INGLES"+OF$
250 LOCATE12,14:PRINT "6   "+RO$+ "DANES"+OF$
260 LOCATE12,16:PRINT "7   "+RO$+ "SUECO"+OF$
270 LOCATE12,18:PRINT "8   "+RO$+ "ITALIANO"+OF$
280 LOCATE9,20:PRINT"┴▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▬"
290 LOCATE9,21:PRINT"█ SELECCIONE ==>    █"
300 LOCATE9,22:PRINT"┐▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▨▬"
310 LOCATE26,21:LINE INPUTAL$::AL=VAL(AL$):IF AL<1 OR AL>8 THEN BEEP:
```

```
      GOTO310
 320 IF AL=1 THENAK=17
 330 IF AL=2 THENAK=10
 340 IF AL=3 THENAK=11
 350 IF AL=4 THENAK=12
 360 IF AL=5 THENAK=13
 370 IF AL=6 THENAK=14
 380 IF AL=7 THENAK=15
 390 IF AL=8 THENAK=16
 400 IF AL<1 OR AL>8 THEN 310
 410 ESC$=CHR$(27):ACE$=CHR$(8)
 420  LPRINT CHR$(27);"R";CHR$(AK);
 430 PR=1:PS=1
 440 CLS:COLOR 1,4:SCREEN0,1
 450 KEY1,"+++++++":KEY2,"^P:Pica":KEY3,"^E:Elit":KEY4,"^C:Comp":
     KEY5,"^F:Enfa":KEY6,"^D:CoDR":KEY7,"^X:Expa":KEY8,"^S:Subr":
     KEY9,"^B:Beep":KEY10,"ESC:Fin"
 460 LI$=""
 470 IK$=INKEY$:IF IK$=""THEN470
 480 IFIK$=CHR$(16) AND PR=1 THEN CL%=80:W=1:GOTO850
 490 IFIK$=CHR$(16) AND PR=2 THEN 840
 500 IFIK$=CHR$(5) AND PR=1 THEN CL%=96:W=2:GOTO870
 510 IFIK$=CHR$(5) AND PR=2 THEN 860
 520 IFIK$=CHR$(3) AND PR=1 THEN CL%=142:W=3:GOTO890
 530 IFIK$=CHR$(3) AND PR=2 THEN 880
 540 IFIK$=CHR$(6) AND PR=1 THEN CL%=80:W=1:GOTO910
 550 IFIK$=CHR$(6) AND PR=2 THEN 900
 560 IFIK$=CHR$(4) AND PR=1 THEN CL%=142:W=3:GOTO930
 570 IFIK$=CHR$(4) AND PR=2 THEN 920
 580 IFIK$=CHR$(24) AND PR=1 THEN CL%=40:W=4:GOTO950
 590 IFIK$=CHR$(24) AND PR=2 THEN 940
 600 IFIK$=CHR$(19) AND PS=1THEN 960
 610 IFIK$=CHR$(19) AND PS>1THEN 980
 620 IFIK$=CHR$(11)ORIK$=CHR$(12)ORIK$=CHR$(18)ORIK$=CHR$(30)ORIK$=
     CHR$(31)ORIK$=CHR$(127)THEN470
 630 IFIK$=CHR$(27)THEN1030
 640 IFIK$=CHR$(9)THENCL%=CL%-8
 650 IFIK$=CHR$(28)THENIK$=" "
 660 IFIK$=CHR$(2)THEN LPRINT ESC$;"8";
 670 IFIK$=CHR$(8)ORIK$=CHR$(29)THEN720
 680 IFIK$=CHR$(39)THEN AC=1:GOTO 770
 690 IFIK$=CHR$(123)THEN DI=1:GOTO 770
 700 IFIK$=CHR$(13)THEN810
 710 GOTO 750
 720 IFLEN(LI$)=0THEN470
 730 LI$=LEFT$(LI$,LEN(LI$)-1)
 740 GOTO 780
 750 IFLEN(LI$)>=CL%THENBEEP: GOTO 470
 760 PR=2
 770 LI$=LI$+IK$
 780 PRINTIK$;:IF AC=1 THEN 990
 790 IF DI=1 THEN1010
 800 GOTO 470
```



```
004A ENT     RDVRM
    ;
    ; Name:           WRTVRM
    ; Function:       Writes to the VRAM addressed by [HL].
    ; Entry:          HL, A
    ; Returns:        None
    ; Modifies:       AF
    ;
004D ENT     WRTVRM
    ;
    ; Name:           SETRD
    ; Function:       Sets up the VDP for read.
    ; Entry:          HL
    ; Returns:        None
    ; Modifies:       AF
    ;
0050 ENT     SETRD
    ;
    ; Name:           SETWRT

    ; Function:       Sets up the VDP for write.
    ; Entry:          HL
    ; Returns:        None
    ; Modifies:       AF
    ;
0053 ENT     SETWRT
    ;
    ; Name:           FILVRM
    ; Function:       Fills the VRAM with the specified data.
    ; Entry:          Address in [HL], length in [BC], data in [Acc]
    ; Returns:        None
    ; Modifies:       AF, BC
    ;
0056 ENT     FILVRM
    ;
    ; Name:           LDIRMV
    ; Function:       Moves a VRAM memory block to memory.
    ; Entry:          Address of source in [HL], destination in [DE],
    ;                 length in [BC].
    ; Returns:        None
    ; Modifies:       All
    ;
0059 ENT     LDIRMV
    ;
    ; Name:           LDIRVM
    ; Function:       Moves block of memory from memory to the VRAM.
    ; Entry:          Address of source in [HL], destination in [DE],
    ;                 length in [BC].
    ; Returns:        None
    ; Modifies:       All
    ;
005C ENT     LDIRVM
    ;
    ; Name:           CHGMOD
    ; Function:       Sets the VDP mode according to SCRMOD.
    ; Entry:          SCRMOD (0..3)
    ; Returns:        None
    ; Modifies:       All
    ;
005F ENT     CHGMOD
    ;
    ; Name:           CHGCLR
    ; Function:       Changes the color of the screen.
    ; Entry:          Foreground color in FORCLR
    ;                 Background color in BAKCLR
    ;                 Border color in BDRCLR
    ; Returns:        None
    ; Modifies:       All
    ;
```

Mapa de Memoria MSX

(Continuación del boletín anterior)

```
COMMENT %

The following routines are used for I/O initialization.

%
;
; Name:          INITIO
; Function:      Performs device initialization.
; Entry:         None
; Returns:       None
; Modifies:      All
;
003B ENT     INITIO
;
; Name:          INIFNK
; Function:      Initializes function key strings.
; Entry:         None
; Returns:       None
; Modifies:      All
;
003E ENT     INIFNK

COMMENT %

The following routines are used to access the VDP (TI9918).

%
;
; Name:          DISSCR
; Function:      Disables screen display.
; Entry:         None
; Returns:       None
; Modifies:      AF, BC
;
0041 ENT     DISSCR
;
; Name:          ENASCR
; Function:      Enables screen display.
; Entry:         None
; Returns:       None
; Modifies:      AF, BC
;
0044 ENT     ENASCR
;
; Name:          WRTVDP
; Function:      Writes to the VDP register.
; Entry:         Register # in [C], data in [B]
; Returns:       None
; Modifies:      AF, BC
;
0047 ENT     WRTVDP
;
; Name:          RDVRM
; Function:      Reads the VRAM addressed by [HL].
; Entry:         HL
; Returns:       A
; Modifies:      AF
;
```



```
810 PRINT:LPRINT LI$:PR=1:GOSUB1100
820 GOTO 460
830 '---subrutina de escritura
840 LPRINT LI$;:CL%=CL%-LEN(LI$)
850 LPRINT ESC$;"!";CHR$(0);:KEY1,"+PICA":GOTO 460
860 LPRINT LI$;:CL%=CL%-LEN(LI$)
870 LPRINT ESC$;"!";CHR$(1);:KEY1,"+ELITE":GOTO 460
880 LPRINT LI$;:CL%=CL%-LEN(LI$)
890 LPRINT ESC$;"!";CHR$(4);:KEY1,"+COMPRI":GOTO 460
900 LPRINT LI$;:CL%=CL%-LEN(LI$)
910 LPRINT ESC$;"!";CHR$(24);:KEY1,"+ENFATI":GOTO 460
920 LPRINT LI$;:CL%=CL%-LEN(LI$)
930 LPRINT ESC$;"!";CHR$(20);:KEY1,"+COM.DR":GOTO 460
940 LPRINT LI$;:CL%=CL%-LEN(LI$)
950 LPRINT ESC$;"!";CHR$(32);:KEY1,"+EXPAND":GOTO 460
960 '=========subrayado
970 LPRINT LI$;:LPRINT ESC$"-";CHR$(1);:PS=PS+1:KEY1,"
    +SUBRAY":CL%=CL%-LEN(LI$): GOTO460
980 LPRINT LI$;:LPRINT ESC$"-";CHR$(0);:PS=1:KEY1,"+      +"
    :CL%=CL%-LEN(LI$):GOTO 460
990 '---------acento
1000 LPRINT LI$;:LPRINT ACE$;:AC=2:CL%=CL%-LEN(LI$)+1:GOTO460
1010 '------dieresis
1020 LPRINT LI$;:LPRINT ACE$;:DI=2:CL%=CL%-LEN(LI$)+1:GOTO460
1030 '---------fin de programa
1040 LPRINT LI$:CLS
1050 LOCATE4,10:PRINT"DESEA ESCRIBIR DE NUEVO  (s/n)";:A$=INPUT$(1)
1060 IF A$="s"ORA$="S" THEN RUN
1070 IF A$="n"ORA$="N" THEN NEW:END
1080 BEEP:GOTO1050
1090 RETURN
1100 '-----long. lineas
1110 IFW=1THEN CL%=80:RETURN
1120 IFW=2THEN CL%=96:RETURN
1130 IFW=3THEN CL%=142:RETURN
1140 IFW=4THEN CL%=40:RETURN
```

BINGO

El juego BINGO está bastante bien planteado y sus gráficos, aunque escuetos, son atractivos y suficientes. Ante un juego de este tipo creemos que la crítica debe ser sin lugar a dudas positiva.

No obstante, hemos observado algunos "puntos flacos":

a) Los efectos de sonido son bastante pobres.

b) Después del BINGO se puede pulsar una tecla y siguen saliendo números.

c) No se comprueba si es posible la línea o el bingo.

A pesar de todo creemos que es muy conveniente publicarlo como está y que los lectores que quieran lo mejoren a su gusto, pues el armazón del programa es válido.

```
10 SCREEN 2
20 COLOR 15,4,5
30 LOCATE 70,80:PRINT"BINGO"
40 FOR J=1 TO 6
50 LINE (70,115)-(180,115),J+1
60 LINE (75,120)-(175,120),J+2
70 LINE (80,130)-(170,130),J+3
80 FOR T=1 TO 150:NEXT T
90 NEXT J
100 FOR T=1 TO 9
110 GOSUB 140
120 NEXT T
130 FOR K=1 TO 900:NEXT K:CLS:GOTO 150
140 FOR F=1 TO200-T^3:NEXT F:BEEP:RETURN
150 GOSUB 690
160 ON KEY GOSUB 580,620,660,660,660
170 KEY (1)ON:KEY(2)ON:KEY(5)ON
180 CIRCLE (175,90),50,1
190 FORA=1TO2STEP.5
200 CIRCLE (175,90),50,1,3.14159*90/180,3.14159*270/180,A
210 CIRCLE (175,90),50,1,3.14159*270/180,3.14159*90/180,A
220 NEXTA
230 FOR H=.3 TO .7 STEP .2
240 CIRCLE (175,90),50,1,3.14159*0/180,3.14159*360/180,H:NEXT H
 250 COLOR 2: LOCATE 120,10:PRINT "precio de linea=";INT(PL):LOCATE 120,20:PRINT
"precio de bingo=";PB
```



```
    DISPLAY (1, 1) ERASE. PERFORM SUBRU.
    DISPLAY (2, 1) 'TECLEAR LOS GRADOS CENTIGRADOS '.
    ACCEPT (2, 33) GRADOS WITH PROMPT.
    MULTIPLY GRADOS BY 8 GIVING RES1.
    DIVIDE RES1 BY 10 GIVING RES2.  MOVE ZEROS TO RES1.
    MULTIPLY GRADOS BY 18 GIVING RES1.
    ADD RES1, 32 TO RES3. MOVE ZEROS TO RES1.
    DIVIDE RES3 BY 10 GIVING RES1.
    ADD GRADOS, 273 TO GRAFAH. DISPLAY (1, 1) ERASE.
    DISPLAY (6, 2) 'CENTIGRADOS'. DISPLAY (6, 25) 'REAMUR'.
    DISPLAY (6, 45) 'FAHRENHEIT'. DISPLAY (6, 65) 'KELVIN'.
    DISPLAY (7, 1) SUB. MOVE GRADOS TO SA.
    DISPLAY (8, 4) SA. MOVE RES2 TO DO. MOVE RES1 TO MA.
    DISPLAY (8, 27) DO. DISPLAY (8, 46) MA.
    DISPLAY (8, 67) GRAFAH.
P.  PERFORM MENU. IF ME="V" GO TO I.
    GO TO P.
FIN.
    STOP RUN.
REAM.
    DISPLAY (1, 1) ERASE. PERFORM SUBRU.
    DISPLAY (2, 1) 'TECLEAR LOS GRADOS REAMUR '.
    ACCEPT (2, 28) GRADOS WITH AUTO-SKIP.
    MULTIPLY GRADOS BY 100 GIVING RES1.
    DIVIDE RES1 BY 80 GIVING RES2.
    MOVE ZEROS TO RES1. DISPLAY (1, 1) ERASE.
    MULTIPLY GRADOS BY 180 GIVING RES1.
    ADD RES1, 32 TO GRAFAH. DIVIDE GRAFAH BY 80 GIVING RES3.
    DISPLAY (6, 8) 'REAMUR'. DISPLAY (6, 30) 'CENTIGRADOS'.
    DISPLAY (6, 65) 'FAHRENHEIT'. DISPLAY (7, 1) SUB.
    DISPLAY (8, 9) GRADOS. DISPLAY (8, 33) RES2.
    DISPLAY (8, 66) RES3.
    PERFORM MENU. IF ME="V" GO TO I. GO TO P.
FAH.

    DISPLAY (1, 1) ERASE. PERFORM SUBRU.
    DISPLAY (2, 1) 'TECLEAR GRADOS FAHRENHEIT '.
    ACCEPT (2, 28) GRADOS WITH AUTO-SKIP.
    SUBTRACT GRADOS FROM 32 GIVING RES1.
    MULTIPLY RES1 BY 100 GIVING RES2.
    DIVIDE RES2 BY 180 GIVING RES3. MOVE ZEROS TO RES1.
    MOVE ZEROS TO RES2. SUBTRACT GRADOS FROM 32 GIVING RES1.
    MULTIPLY RES1 BY 80 GIVING RES2.
    DIVIDE RES2 BY 180 GIVING GRAFAH. DISPLAY (1, 1) ERASE.
    DISPLAY (6, 8) 'FAHRENHEIT'. DISPLAY (6, 30) 'CENTIGRADOS'.
    DISPLAY (6, 55) 'REAMUR'. DISPLAY (7, 1) SUB.
    DISPLAY (8, 9) GRADOS. DISPLAY (8, 33) RES3.
    DISPLAY (8, 57) GRAFAH. PERFORM MENU. IF ME="V" GO TO I.
    GO TO F.
MENU.
    DISPLAY (21, 1) 'Para volver al menu pulsar  V '.
    ACCEPT (21, 30) ME WITH AUTO-SKIP.
SUBRU.
    MOVE ZEROS TO RES1. MOVE ZEROS TO RES2. MOVE ZEROS TO RES3.
    MOVE ZEROS TO GRADOS. MOVE ZEROS TO GRAFAH.
    MOVE ZERO TO OPCION. MOVE SPACE TO ME.
    MOVE ZEROS TO SA. MOVE ZEROS TO DO. MOVE ZEROS TO LU.
    MOVE ZEROS TO MA.
```

```
        3: BEGIN
              CLRSCR; GOTOXY(3,5); WRITE('TECLEAR GRADOS FAHRENHEIT ');
              READLN(GRADOS); CLRSCR; RES1:=(GRADOS-32)*100/180;
              RES2:=(GRADOS-32)*80/180; GOTOXY(5,5); WRITE('FAHRENHEIT');
              GOTOXY(30,5); WRITE('CENTIGRADOS'); GOTOXY(60,5);
              WRITELN('REAMUR'); GOTOXY(1,6);
--------');   WRITELN('-----------------------------------------------------------------
              GOTOXY(7,7); WRITE(GRADOS:4:2); GOTOXY(33,7);
              WRITE(RES1:4:2); GOTOXY(62,7); WRITE(RES2:4:2); GOTO 200;
           END;
        4: END
    END.
```

COBOL

```
TPYE TYPE MARY.COB
       IDENTIFICATION DIVISION.
       PROGRAM-ID. EJEMPLO1.
       AUTHOR. VICENTE GOMEZ MARTINEZ.
       DATE-WRITTEN. 20-07-1986.
       DATE-COMPILED. 28-07-1986.
       ENVIRONMENT DIVISION.
       CONFIGURATION SECTION.
       SOURCE-COMPUTER. SPECTRAVIDEO.
       OBJECT-COMPUTER. SPECTRAVIDEO.
       SPECIAL-NAMES.
           DECIMAL-POINT IS COMMA.
       DATA DIVISION.
       WORKING-STORAGE SECTION.
       77  GRADOS PIC S99V99 VALUE ZEROS.
       77  RES1 PIC S99V99 VALUE ZEROS.
       77  RES2 PIC S99V99 VALUE ZEROS.
       77  RES3 PIC S99V99 VALUE ZEROS.
       77  SUB PIC X(79) VALUE ALL "=".
       77  ME PIC A VALUE SPACE.
       77  GRAFAH PIC S99V99 VALUE ZEROS.
       77  OPCION PIC 9 VALUE ZERO.
       01  CAMPOS-EDICION.
           02 SA PIC -ZZ,ZZ.
           02 DO PIC -ZZ,ZZ.
           02 LU PIC -ZZ,ZZ.
           02 MA PIC -ZZ,ZZ.
       PROCEDURE DIVISION.
       I.  DISPLAY (1, 1) ERASE.
       C.  DISPLAY (4, 30) 'M E N U'.
           DISPLAY (7, 18) '1.- Pasar de centigrados a demas escalas'.
           DISPLAY (9, 18) '2.- Pasar de Reamur a las demas escalas'.
           DISPLAY (11, 18) '3.- Pasar de Fahrenheit    "     "    '.
           DISPLAY (13, 18) '4.- Fin'.
       D.  DISPLAY (20, 7) 'Opcion '.
           ACCEPT (20, 14) OPCION WITH AUTO-SKIP.
           IF (OPCION<1) OR (OPCION>4) GO TO D.
           GO TO CENTI, REAM, FAH, FIN DEPENDING ON OPCION.
       CENTI.
```



```
260 LINE(175-50,90)-(175-50-16,175),15,BF:LINE(175+50,90)-(175+65,175),15,BF
270 LINE (2,1)-(100,140),2,BF
280 FOR H=1 TO 8
290 READ A$
300 P$=P$+CHR$(VAL("&B"+A$)): NEXT H
310 SPRITE$(1)=P$
320 GOTO330
330 LINE (45,160)-(30,180),1,BF:COLOR15,4,5
340 SOUND 7,RND(255):SOUND 8,9:FOR T=1 TO 3
350 FOR G=125TO 40STEP -21
360 H=INT(RND(TIME)*360)
370 A=165+RND(2)*40*SIN(3.14159*H/180)
380 B=90+RND(2)*40*COS(3.14159*H/180)
390 PUT SPRITE RND(2)*3,(A,B),2,1
400 PUT SPRITE 30,(168.G),1,1
410 NEXT G
420 NEXTT
430 N=RND(-TIME)
440 SOUND 8,11
450 S=INT(RND(6)*99)+1
460 IF A(S)=1 THEN 430
470 A(S)=1
480 TIME =0
490 LOCATE52,167:PRINTS:Z=0:P=9:F=0:C=0:COLOR 1,3
500 FOR K=1 TO 50:NEXTK
510 FOR F=2 TO 128 STEP 8
520 IF Z+1>9 THEN P=15
530 FOR C=2 TO 82 STEP P
540 Z=Z+1
550 IF S=Z THEN LOCATE C,F:PRINTZ:GOSUB 670:GOTO 330
560 NEXT C
570 NEXT F
580 LOCATE 155,170:COLOR 1:PRINT"lineaaaa.."
590 SOUND 8,15
600 A$=INKEY$:IF A$<>" " THEN GOTO 600
610 RETURN
620 LOCATE 155,180:COLOR 1:PRINT"BINGOOO..
630 SOUND 8,15
640 A$=INKEY$:IF A$<>" "THEN GOTO 640
650 RETURN
660 RUN
670 IF TIME >150 THEN RETURN
680 GOTO 670
690 COLOR 4,11,15
700 SCREEN 0,0
710 LOCATE 14,1:POKE 65077',1:PRINT"-bingo-":POKE 65077!,0
720 LOCATE 5,5:INPUT "Numero de Jugadores(1-100)";A
730 IF A>100 OR A<1 OR A<>INT(A)THEN 820
740 LOCATE 5,8:INPUT "Precio por carton";B
750 MA=A*B
760 PL=MA*20/100:PB=MA-PL
770 LOCATE 5,11:PRINT"precio por linea=";INT(PL)
780 LOCATE 5,14:PRINT"precio al bingo=";PB
790 LOCATE 7,18,0:PRINT"F1-linea;F2-bingo;F5-empezar"
```

```
800 LOCATE 7,20:PRINT"space-retorno al juego"
810 A$=INKEY$:IF A$="" THEN 810ELSE SCREEN1,1: COLOR 15,4,5:DIM A(99):RETURN
820 LOCATE 3,22:PRINT" NO SABES LEER O GUE?":FOR J=1 TO 400:NEXT J:LOCATE 3,22:P
RINT"                                ":GOTO 720
830 DATA 00111100
840 DATA 01111110
850 DATA 10000001
860 DATA 10000001
870 DATA 10000001
880 DATA 10000001
890 DATA 01111110
900 DATA 00111100
```

POLIGONOS REGULARES

Para aprender trucos gráficos

```
10 INPUT "dime el numero de lados";N
20 SCREEN 1
25 PSET (120+40*COS(3.14159*0/180),80
   +40*SIN(3.14159*0/180)),2
30 FOR T=0 TO 360 STEP 360/N
40 LET C=120+40*COS(3.14159*T/180)
50 LET S=80+40*SIN(3.14159*T/180)
60 LINE-(C,S),2
70 NEXT T
75 PAINT(120,80),2
80 A$=INKEY$:IF A$="" THEN 80
90 CLS:GOTO 10
100 END
```


El mismo programa en dos lenguajes ..

PASCAL

```
A>TYPE TEMPER.PAS
PROGRAM Temperatura;

LABEL 100, 200;

VAR
   OPCION:INTEGER;
   RESP:CHAR;
   GRADOS,RES1,RES2,RES3:REAL;
BEGIN
100 :CLRSCR; GOTOXY(29,5);
      WRITELN('M E N U'); GOTOXY(20,7);
      WRITELN('1- DE CENTIGRADOS A DEMAS ESCALAS');
      GOTOXY(20,9); WRITELN('2- DE REAMUR A DEMAS ESCALAS');
      GOTOXY(20,11); WRITELN('3- DE FAHRENHEIT A DEMAS ESCALAS');
      GOTOXY(20,13); WRITELN('4- FIN DE TRABAJOS');
      GOTOXY(2,21); WRITE ('OPCION '); READLN(OPCION);
      IF (OPCION<1) OR (OPCION>4) THEN GOTO 100;
      CASE OPCION OF
           1: BEGIN
                  CLRSCR; GOTOXY(3,5); WRITE('TECLEAR GRADOS CENTIGRADOS ');
                  READLN(GRADOS); CLRSCR;
                  RES1:=(GRADOS*80)/100; RES2:=((GRADOS*180)+32)/100;
                  RES3:=GRADOS+273;
                  GOTOXY(3,5); WRITE('CENTIGRADOS');
                  GOTOXY(22,5); WRITE('REAMUR'); GOTOXY(40,5);
                  WRITE('FAHRENHEIT'); GOTOXY(65,5);
                  WRITELN('KELVIN'); GOTOXY(1,6);
-------');        WRITELN('-------------------------------------------------------------------------
                  GOTOXY(5,7); WRITE(GRADOS:4:2); GOTOXY(24,7); WRITE(RES1:4:3);

(* PARA ESCRIBIR DECIMALES SE PONE LA VARIABLE A ESCRIBIR Y LUEGO SEPARADOS
   POR DOS PUNTOS LA LONGITUD DE LA PARTE ENTERA Y LA LONGITUD DE LA DECIMAL *)

(* EJEMPLO  GRADOS:4:2                                                      *)

                  GOTOXY(43,7); WRITE(RES2:4:2); GOTOXY(67,7); WRITE(RES3:4:2);
             200: GOTOXY(2,20); WRITE('PARA VOLVER AL MENU PULSAR  V ');
                  READLN(RESP); IF RESP='V' THEN GOTO 100 ELSE GOTO 200;
             END;
          2: BEGIN
                  CLRSCR; GOTOXY(3,5); WRITE('TECLEAR GRADOS REAMUR ');
                  READLN(GRADOS); CLRSCR; RES1:=(GRADOS*100)/80;
                  RES2:=(GRADOS*180)+32/80; GOTOXY(5,5); WRITE('REAMUR');
                  GOTOXY(30,5); WRITE('CENTIGRADOS ); GOTOXY(60,5);
                  WRITE('FAHRENHEIT'); GOTOXY(1,6);
-------');        WRITELN('-------------------------------------------------------------------------
                  GOTOXY(7,7); WRITE(GRADOS:4:2); GOTOXY(33,7); WRITE(RES1:4:2);
                  GOTOXY(62,7); WRITE(RES2:4:2); GOTO 200
             END;
```